

# **THE 2ND NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE AND INFORMATION TECHNOLOGIES**



## **NCAIIT'25**

- **ABDELMADJID BENMACHICHE**
- **ALI ABDELLATIF BETOUIL**
- **CHAOUKI CHEMAM**
- **KHADIJA RAIS**



**CHADLI BENDJEDID  
UNIVERSITY**



**6 MAY 2025**

NCAIIT'25 Conference Presentations

**Presentation Title**

A Cycle-Based Hub Location Approach For Pharmaceutical Supply Chain

A Semantic User Profile with Machine Learning for Social Network Communities

A Study of Aggregation Strategies for Ensemble Multi-Label Classifiers

Adaptive Real-Time Scheduling Algorithms for Embedded Systems

Advancements in Recommender Systems for Personalized Learning: A Comprehensive Survey

Advancing a Sustainable and Adaptive Energy Future: Enabling Renewable Integration for Intelligent Power Management

AraBERT-Based Contextual Model for Quranic Verse Classification

Attention U-NET for Skin Cancer Segmentation

Bridging AI and Microbiology: Deep Learning for Identifying Eugenol-Derived Antimicrobial Molecules

Data Injection into Autonomous Drones: Challenges, Risks and Solutions Based on Artificial Intelligence

Deep Learning for Cybersecurity: CNN-Based Intrusion Detection

DeepPNDM : A Deep Learning Model for Permanent Neonatal Diabetes Mellitus Diagnosis

Developing and Evaluating Lightweight Cryptographic Algorithms for Secure Embedded Systems in IoT Devices

Advanced Machine Learning and Swarm Intelligence for Enhanced Cyanobacterial Bloom Prediction: A Comparative Study

Enhancing the Security and Privacy Protection of Video Surveillance Data in Smart Cities Using Digital Watermarking and Smart Contracts

Ensemble Learning vs Single Deep Learning Models for Blood Cancer Classification: A Comparative Study

Explainable Deep Learning for Coronary Heart Disease Prediction: A Framingham Dataset Approach

Exploring the Application of Federated Learning in Cybersecurity for Enhanced Threat Detection

Exploring the Conceptual Framework of Artificial Intelligence: From foundations to Ethical Implications

Harnessing TinyML for Efficient AI and IoT-Based Self-Adaptive Software Systems

Heart Disease Prediction Using Machine Learning Models: A Comparative Study

Hybrid Adversarial Machine Learning for Robust Cybersecurity

Hybrid IDS Using Signature-Based and Anomaly-Based Detection

An interference-aware multipath routing algorithm for WSDNs

---

---

**Presentation Title**

---

Intrusion Detection in Network Traffic Using Random Forest Algorithms

---

Investigate the degradation of marine propulsion systems under predictive maintenance: Comparison of Machine Learning Techniques

---

IoT Security Using Blockchain and AI

---

KT-GNC: A Novel AI Framework for Smart Traffic Congestion Avoidance

---

Large Language Models for Health Recommender Systems: A Review

---

Large-Scale E-Commerce Product Selection Using Skyline Queries in Heterogeneous Computing Environments

---

Laryngitis Detection System Using Machine Learning Models

---

Mining sequential patterns with quantities under constraints

---

Mobile Edge Computing Architecture for Network Management and Security

---

Modern Deep Learning Techniques for 3D Facial Reconstruction

---

Network Traffic Control System for Mobile Video Streaming

---

Non-Invasive Dysphonia Detection Using Acoustic Features and k-Nearest Neighbors

---

On Sensitive Content Integrity Techniques : A Short Review

---

Optimization of Deep Learning Models for Embedded Vision Systems

---

Predicting Client Subscription to Term Deposits Using Machine Learning

---

Quantum Privacy in Secure Medical Systems

---

Real-Time Image Processing Algorithms for Embedded Systems

---

Real-Time Machine Learning for Embedded Anomaly Detection

---

Secure and Private Network to Mitigate Real Estate Frauds in Algeria

---

Secure Video Transmission Via UDP and Blockchain for Smart Cities

---

Synthesizing Brain Images with GANs: A Compact Review of Methods and Metrics

---

Towards a system for Detection, Prevention and Resilience against Data Injection Attacks on Autonomous Drones

---

---

## NCAIIT'25 Conference Papers

Presentation Title	Authors
A Cycle-Based Hub Location Approach For Pharmaceutical Supply Chain	O. Kemmar and A. Kemmar
A Semantic User Profile with Machine Learning for Social Network Communities	S. Bousalem, M. Chelghoum, H. Guergour and K. Zarour
A Study of Aggregation Strategies for Ensemble Multi-Label Classifiers	S. Guehria and F. Kherissi
Adaptive Real-Time Scheduling Algorithms for Embedded Systems	A. Benmachiche, K. Rais and H. Slimi
Advancing a Sustainable and Adaptive Energy Future: Enabling Renewable Integration for Intelligent Power Management	M. S. Benkhalfallah and S. Kouah
Attention U-NET for Skin Cancer Segmentation	M. Chibani
Bridging AI and Microbiology: Deep Learning for Identifying Eugenol-Derived Antimicrobial Molecules	C. Touati, A. Kemmar and F. Chaib
Data Injection into Autonomous Drones: Challenges, Risks and Solutions Based on Artificial Intelligence	S. Maalem, A. Bouamrane, M. S. Kahil and M. Derdour
Developing and Evaluating Lightweight Cryptographic Algorithms for Secure Embedded Systems in IoT Devices	B. Sedraoui and A. Benmachiche
Advanced Machine Learning and Swarm Intelligence for Enhanced Cyanobacterial Bloom Prediction: A Comparative Study	O. Bounekhla, M. Hemici, N. Dendani, A. Saoudi and N. Azizi
Enhancing the Security and Privacy Protection of Video Surveillance Data in Smart Cities Using Digital Watermarking and Smart Contracts	M. Kheraifia, A. Sahraoui, S. Maalem and M. Derdour
Explainable Deep Learning for Coronary Heart Disease Prediction: A Framingham Dataset Approach	N. Menaceur, S. Kouah and M. Derdour
Exploring the Application of Federated Learning in Cybersecurity for Enhanced Threat Detection	I. Soualmia, A. Benmachiche, S. Maalem and I. Boutabia
Exploring the Conceptual Framework of Artificial Intelligence: From foundations to Ethical Implications	Sara Gasmi, Safa Gasmi and Bouhadada Tahar
Heart Disease Prediction Using Machine Learning Models: A Comparative Study	Sara Gasmi, Safa Gasmi and A. Djebbar
Hybrid Adversarial Machine Learning for Robust Cybersecurity	R. Mounira, A. Benmachiche and M. Majda

Presentation Title	Authors
Hybrid IDS Using Signature-Based and Anomaly-Based Detection	M. Boutassetta, A. Makhoul, N. Mes-saoudi, A. Benùachiche and I. Boutabia
IoT Security Using Blockchain and AI	A. Djaber, A. Ben-machiche and M. Der-dour
Large-Scale E-Commerce Product Selection Using Skyline Queries in Heterogeneous Computing Environments	W. Khemes
Mining sequential patterns with quantities under constraints	A. Kemmar, A. Djeb-bar, C. Touati and O. Kemmar
Mobile Edge Computing Architecture for Network Management and Security	A. Cheriet, A. Sahraoui, S. Maalem and M. Der-dour
Modern Deep Learning Techniques for 3D Facial Reconstruction	R. Agaba, M. Malah and F. Abbas
Network Traffic Control System for Mobile Video Streaming	A. Cheriet, A. Sahraoui, S. Maalem and M. Der-dour
Optimization of Deep Learning Models for Embedded Vision Sys-tems	I. Boutabia, A. Ben-machiche and A. Betouil
Predicting Client Subscription to Term Deposits Using Machine Learning	A. Djebbar and A. Kemmar
Quantum Privacy in Secure Medical Systems	H. Slimi, M. Dourdour, K. Sadek, A. Bouam-rane, S. Abdelatif
Real-Time Image Processing Algorithms for Embedded Systems	S. BOUFAIDA, A. BENMACHICHE and M. MAATALLAH
Real-Time Machine Learning for Embedded Anomaly Detection	A. Benmachiche, K. Rais and H. Slimi
Secure Video Transmission Via UDP and Blockchain for Smart Cities	M. Kheraifia, A. Sahraoui, S. Maalem and M. Dourdour3
Synthesizing Brain Images with GANs: A Compact Review of Meth-ods and Metrics	K. Rais, M. Amroune and M. Y. Haouam
Towards a system for Detection, Prevention and Resilience against Data Injection Attacks on Autonomous Drones	M. Kahil, H. Slimi, S. Maalem, A. Bouamrane and M. Dourdour

# A Cycle-Based Hub Location Approach For Pharmaceutical Supply Chain

Omar Kemmar<sup>1</sup>, Amina Kemmar<sup>2</sup>

<sup>1</sup>University of Relizane - Ahmed Zabana, Relizane, Algeria

LIO, University of Oran 1 Ahmed Ben Bella, Oran, Algeria

omar.kemmar@univ-relizane.dz

<sup>2</sup>Oran Graduate School of Economics, BP 65 CH 2 Achaba Hnifi - USTO, Oran, Algeria

LITIO, University of Oran 1 Ahmed Ben Bella, Oran, Algeria

kemmar.amina@gmail.com

**Abstract**—The efficient distribution of pharmaceutical products is critical to ensuring timely patient access while minimizing logistics costs. Traditional hub-and-spoke models often suffer from high setup costs, inefficient delivery routes, and inflexible hub placement. In this work, we propose a cycle-based hub location model where pharmacies form local clusters (cycles) before receiving supplies from central hubs. This structure optimizes distribution by reducing long-distance transportation costs and leveraging local inventory exchanges. To solve this problem, we introduce a mixed integer linear programming mathematical model with an exponential number of constraints and a metaheuristic/hyper-heuristic optimization framework to solve large-scale instances. Unlike classical approaches, our model does not consider time constraints or capacity limitations, focusing purely on cost efficiency. The results demonstrate that the proposed approaches significantly reduce logistics expenses while maintaining high service levels.

**Index Terms**—Hub Location Problem, Pharmaceutical Supply Chain, Cost Optimization, Metaheuristics, Hyper-Heuristic

## I. INTRODUCTION

The **Hub Location Problem (HLP)** is a strategic optimization problem focused on determining the optimal locations for hubs and assigning demand points to these hubs to minimize costs or maximize efficiency in transportation, communication, and logistics networks [O'Kelly \(1987\)](#). Hubs act as central consolidation and redistribution points, leveraging economies of scale to reduce transportation costs between origin-destination pairs. Key components include nodes (locations), hubs (central points), flows (goods, passengers, or data), and cost structures (fixed hub costs, transportation costs, and inter-hub discounts). Variants of HLP include single allocation (each node assigned to one hub) vs. multiple allocation (nodes assigned to multiple hubs), uncapacitated vs. capacitated hubs, p-hub median (fixed number of hubs) [Campbell \(1994\)](#), hub covering (distance constraints), competitive (multiple firms), dynamic (changing demand), stochastic (uncertainty), and hierarchical (multiple hub levels) [Alumur and Kara \(2008\)](#). Applications span industries such

as transportation (airline and freight networks), telecommunications (data centers), postal services (sorting centers), public services (emergency response), supply chain management (distribution networks), energy distribution, and disaster relief [Farahani et al. \(2013\)](#). HLP is typically formulated using mathematical optimization models, often solved via integer or mixed-integer programming, and remains a critical tool for designing efficient and cost-effective networks [Contreras and Fernández \(2012\)](#) (see [Figure 1](#)).

Pharmaceutical supply chains are highly sensitive networks where timely and cost-effective distribution is crucial. Pharmacies must ensure they have sufficient stock of frequently demanded medications while also having the ability to rapidly obtain non-stocked prescriptions. The distribution of these medications is typically handled via a hub-and-spoke model, where central warehouses supply multiple pharmacies within a region.

However, this traditional model presents several challenges:

- High setup and operational costs for hubs.
- Inefficient direct-to-pharmacy deliveries, leading to increased transportation costs.
- Rigid hub locations, which do not adapt well to fluctuating demand patterns.

To address these limitations, we propose a cycle-based hub location model in [Figure 2](#). Instead of direct hub-to-pharmacy deliveries, pharmacies within a region form local cycles, allowing them to redistribute common medications before relying on the central hub. This structure reduces overall costs by minimizing unnecessary hub-to-pharmacy shipments and leveraging local inventory exchanges.

### A. Literature Review

This study can be viewed as a variation of the uncapacitated single allocation hub location problem (USAHLP). Below, we examine some closely related works, focusing primarily on the pharmaceutical supply

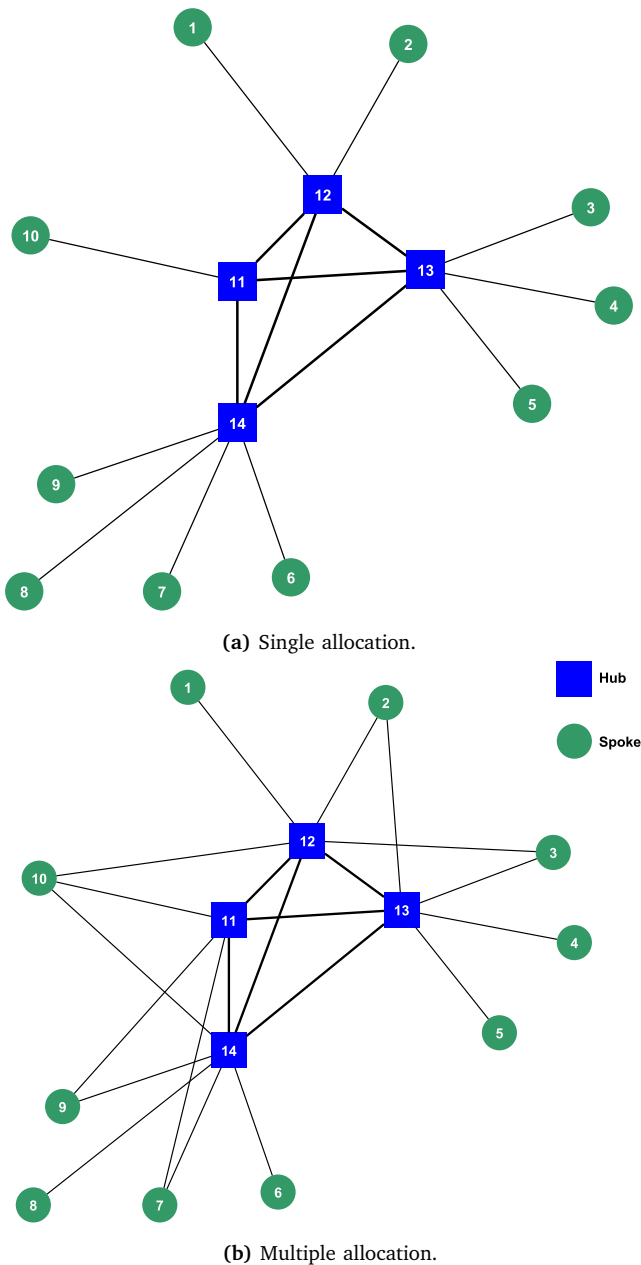


Fig. 1: A typical hub-and-spoke networks.

chain and hub location problems in logistics.

1) *Pharmaceutical Supply Chain*: The optimization of pharmaceutical supply chains has been a critical area of research, particularly in cold-chain logistics. Dongjiu et al. (2012) proposed a two-stage model to optimize a hub-and-spoke network, using the Improved Analytic Hierarchy Process (IAHP) to identify distribution centers and improve consolidation and delivery processes. Their work highlights the role of efficient transportation networks in reducing costs and enhancing logistics effectiveness, leveraging mathematical graph theory and collection-delivery models to boost economic benefits and market

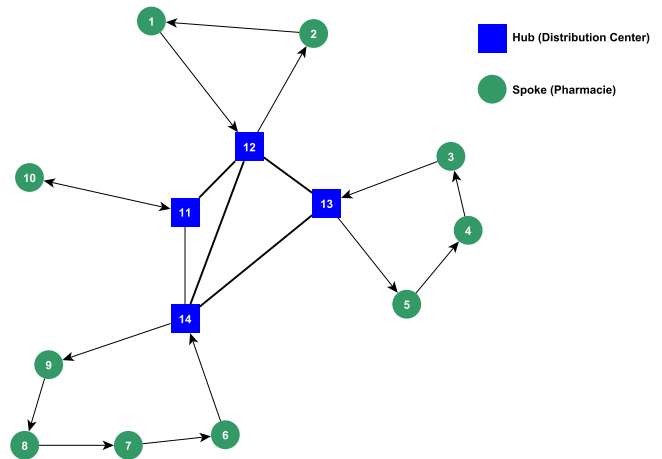


Fig. 2: A Cycle-Based Hub and Spoke network

competitiveness in the pharmaceutical cold-chain sector.

In the context of developing countries, Haial et al. (2016) introduced a framework for designing transportation networks in pharmaceutical supply chains, focusing on Morocco. Their approach involves analyzing the current network, optimizing distribution through location-allocation decisions, and selecting a transportation strategy using the TOPSIS method. The study emphasizes minimizing costs while maximizing service levels, ultimately advocating for a milk-run strategy to improve efficiency in resource-constrained settings.

Recent research by AlZaidan et al. (2024) focuses on optimizing Qatar's pharmaceutical supply chain using a Mixed-Integer Programming (MIP) model to determine optimal warehouse locations. Their findings suggest that a centralized warehouse in the West region minimizes transportation costs and enhances distribution efficiency, addressing gaps in the literature and encouraging future research on integrating qualitative factors and alternative optimization approaches.

Building on this, Benabbou et al. (2020) presented a framework for redesigning pharmaceutical transportation networks, with a case study in Morocco. Their framework includes network configuration, location-allocation optimization, and transportation strategy selection using Multi-Criteria Decision Analysis (MCDA). The study highlights the milk-run method as an effective strategy for cost-efficient and timely pharmaceutical delivery, with future research focusing on advanced mathematical modeling and strategy refinement.

In the broader context of supply chain innovation, Lucchese et al. (2020) explored advancements in Industry 4.0, emphasizing the integration of smart technologies in manufacturing. While their work provides valuable insights into technical methodologies and future industrial applications, some sections suffer from character recognition issues, limiting comprehensibility.

To address uncertainties in pharmaceutical supply chains, Nozari and Rahmaty (2023) introduced a two-stage probabilistic programming model that combines

the Whale Optimization Algorithm and Genetic Algorithm (WOGA). Their results demonstrate that WOGA outperforms traditional algorithms in minimizing costs and addressing demand uncertainties, offering significant potential for cost reduction and timely pharmaceutical delivery.

The challenges of multi-center logistics were tackled by [Yuan and Gao \(2022\)](#), who proposed a hybrid algorithm for optimizing pharmaceutical distribution routes. Their model, which combines fuzzy C-means, sequential quadratic programming, and variable neighborhood search, effectively reduces distribution risks and costs, emphasizing the importance of addressing dynamic uncertainties in pharmaceutical logistics.

In response to disruptions like the COVID-19 pandemic, [Kochakkashani et al. \(2024\)](#) developed a Mixed-Integer Nonlinear Programming (MINLP) model to enhance pharmaceutical supply chain resilience. Their framework integrates unsupervised learning algorithms and a joint chance constraint formulation to manage inventory and address demand uncertainties, ensuring reliable supply during crises while emphasizing equity in distribution.

Sustainability in pharmaceutical supply chains has also gained attention. [Low et al. \(2016\)](#) proposed a framework for designing environmentally sustainable networks, using the Analytic Hierarchy Process (AHP) for site selection and evaluating economic and environmental performance. Their case study demonstrates the framework's effectiveness in reducing greenhouse gas emissions and improving sustainability.

Further advancing this field, [Nasrollahi and Razmi \(2021\)](#) developed a multi-objective model for pharmaceutical supply chains, aiming to maximize demand coverage while minimizing costs. Their case study in Iran shows that considering varying reliability levels for pharmaceutical substances improves demand satisfaction without significantly increasing costs, enhancing the practical reliability of supply chains.

The broader challenges of supply chain optimization were explored by [Garcia and You \(2015\)](#), who emphasized the need for advanced algorithms to address multi-scale, multi-objective, and multi-player complexities. Their work highlights the importance of integrating sustainability into supply chain design and leveraging disruptive technologies like Big Data.

Equitable access to pharmaceuticals was the focus of [Duarte et al. \(2022\)](#), who developed a decision support tool to optimize vaccine distribution based on disease burden metrics. Their Multi-Objective Mixed Integer Linear Programming (MILP) model emphasizes sustainability and accessibility, advocating for prioritizing high-burden regions to achieve equitable access.

Inventory management in pharmaceutical supply chains was addressed by [Ward \(2017\)](#), who explored the benefits of centralizing inventory through hub-and-spoke and national network designs. Their findings suggest significant cost savings, particularly for products with sporadic de-

mand, and recommend further research on incorporating additional constraints.

Finally, [Potters et al. \(2024\)](#) proposed a Mixed-Integer Linear Programming (MILP) model to optimize last-mile logistics in pharmaceutical supply chains, focusing on medication synchronization and diverse delivery modes. Their case study demonstrates a 34% improvement in financial outcomes, highlighting the importance of synchronization in cold supply chains.

Various studies have focused on optimizing pharmaceutical logistics by addressing key areas such as inventory management at pharmacies, supplier selection and hub placement, and dynamic routing for medical deliveries. However, most existing research primarily emphasizes traditional hub-and-spoke models and often overlooks the potential benefits of cycle-based redistribution strategies.

2) *Hub Location Problems in Logistics:* [Neamatian Monemi et al. \(2021\)](#) presented a case study of humanitarian aid distribution. They proposed a Multiperiod Hub Location Problem with Serial Demand (MPHLPD) with a Mixed Integer Programming (MIP) formulation. Several valid inequalities were identified, and the benders decomposition approach was used as a solving method for large instances.

A new hub location problem structure was proposed in [Kemmar et al. \(2021\)](#), where the term runaway node was used for the first time as far as we know. In this work, they proposed a MIP formulation in addition to hyper-heuristic and variable neighborhood search approaches.

[Danach et al. \(2019\)](#) tackled the single allocation hub location and routing problem. The volume of flow passing by a spoke-level edge must not exceed the capacity of transporters available on it. Three methods are proposed as solutions: a MIP formulation, a Lagrangian relaxation, and a hyper-heuristic.

[Azizi \(2019\)](#) studied the uncapacitated Single Allocation p-Hub Location Problem under hub disruption risk. A MIP formulation and a Particle Swarm Optimisation (PSO) algorithm are proposed where the author constructs networks in which a backup hub is assigned to every single demand to be served in case of disruption.

The reliable single allocation Hub Location Problem was introduced in [Rostami et al. \(2018\)](#). A two-stage formulation is given with the benders decomposition approach to solve large-scale instances. In this work, when a hub breaks down, its corresponding flow is rerouted via a single backup hub node.

[Zhong et al. \(2018\)](#) studied the hierarchical hub location model and proposed a MIP formulation with hub capacity constraints as well as a hybrid meta-heuristic (tabu search and genetic algorithm).

[Gelareh et al. \(2017a\)](#) and [Gelareh et al. \(2017b\)](#) tackled the Bounded Cardinality Capacitated Hub Routing Problem (BCCHRP) with route capacity constraints. A 2-

index MIP formulation and a branch-and-cut approach based on Benders decomposition are proposed.

A MIP formulation and a branch-and-cut algorithm for the Cycle Hub Location Problem (CHLP) are presented in Contreras et al. (2016). A greedy randomized adaptive search procedure (GRASP) has been developed to solve large-scale instances of the CHLP.

In Chaharsooghi et al. (2016), a two-stage stochastic model and an adaptive large neighborhood search meta-heuristic approach were given for the reliable uncapacitated multiple allocation hub location problem under hub disruptions. In this context, the spokes allocated to a failing hub are either reallocated to other still working hubs, or a penalty is paid in the case that they do not receive any service due to the high reallocation costs.

Mohammadi et al. (2016) tackles the single allocation  $p$ -hub center-median problem under data uncertainty. A bi-objective mixed-integer non-linear programming model and an evolutionary algorithm were proposed. Interested readers on reliability network problems are also referred to the following studies: Yahyaie et al. (2019), Zhalechian et al. (2018), Cardoso et al. (2015).

In de Sá et al. (2015a) studied the Hub Line Location Problem (HLLP) for public transportation systems introduced in de Sá et al. (2015b). They introduced a benders decomposition algorithm and several metaheuristics for the problem.

In Rodriguez-Martin et al. (2014), the hub location and routing problem were considered, where the number of hubs is exogenous information about the problem, and the capacity constraint is set in terms of the number of spokes per cycle. A MIP formulation and a branch-and-cut algorithm were proposed for the studied problem.

Gelareh et al. (2013) proposed a hub-and-spoke structure with one central hub cycle and spoke-level (feeder) cycles attached to every hub node. They used a Lagrangian decomposition approach equipped with a Lagrangian heuristic to solve the problem.

Yang et al. (2013) tackled the  $p$ -Hub Center Problem in fuzzy environments. A hybrid particle swarm optimization (PSO) algorithm was proposed based on the combination of PSO, genetic operators, and local search (LS).

Alumur et al. (2012) studied the hierarchical multimodal hub location problem. A MIP formulation, where two sorts of hub nodes and hub edges are considered (for ground and air transportation), with a time-definite delivery service.

Kim and O'Kelly (2009) introduced a reliable  $P$ -Hub Location Problem for telecommunication networks where single and multiple assignment schemes are considered.

Yaman et al. (2007) proposed a minimax mathematical model for the latest arrival hub location problem in ground-based cargo delivery systems with stopovers.

For the capacitated single allocation hub location problem (CSAHL), Carello et al. (2004) proposed a local

search approach and different meta-heuristic algorithms where the hubs are transit nodes and the spokes are access nodes.

O'Kelly (1992) introduced the single allocation hub location problem (SAHLP). The author proposed a quadratic integer formulation where the number of hubs was an endogenous part of the problem with fixed costs. Campbell (1992) introduced the first model for the multiple allocation problem.

Most studies on Hub Location Problems in the literature focus on traditional hub-and-spoke network structures. Additionally, the majority of non-exact approaches proposed in previous research rely on heuristics and meta-heuristics, while hyper-heuristic methods remain largely unexplored in this context, particularly in pharmaceutical supply chains. The importance of optimizing pharmaceutical supply chains has gained significant attention, especially after the COVID-19 pandemic, highlighting the need for more resilient and efficient distribution models.

The main features of the most related contributions in the literature are summarized in Table I.

## B. Contribution and scope

The problem addressed in this paper is inspired by real-world challenges in pharmaceutical supply chains, where traditional hub-and-spoke networks often lead to inefficiencies in cost, excessive dependency on hubs, and vulnerability to disruptions. In practice, pharmaceutical distribution networks rely heavily on centralized hubs, which can create bottlenecks and increase the risk of failures due to supply chain disruptions, strikes, or unexpected demand fluctuations. These challenges highlight the need for a more flexible and cost-effective distribution structure.

To overcome these limitations, we propose a cycle-based hub-and-spoke model that enhances redundancy and efficiency by allowing pharmacies, which act as spokes in the network, to form local redistribution cycles before relying on hubs. This structure introduces pharmacy-to-pharmacy exchanges, providing an alternative to direct hub dependency. These inter-pharmacy transfers function as cost-effective redistribution links, reducing the need for additional backup hubs while ensuring continuous supply in case of hub failures. Unlike traditional backup hubs, which are costly to establish and maintain, the cycle-based structure leverages local redistribution links that require minimal additional infrastructure.

The decentralized redundancy approach embedded in this model ensures that even when central hubs become temporarily unavailable, pharmacy cycles can redistribute supplies locally, acting as a buffer against disruptions. By integrating cycle-based redundancy, the proposed model represents a significant enhancement to classical hub-and-spoke systems, leading to improved cost efficiency, service reliability, and adaptability in pharmaceutical logistics.

Work	Alloc.	Num. hubs	Objective	Capacity	Solution method
Current work	SA	Exogenous	Cost	No	MIP + Meta-heuristic + Hyper-heuristic
Danach et al. (2019)	SA	Exogenous	Time (transit transshipment) +	Yes	MIP + Lagrangian relaxation + Hyper-heuristic
Zhong et al. (2018)	SA	Endogenous	Cost	Yes	MIP + Meta-heuristic
Contreras et al. (2016)	SA	Exogenous	Cost	No	MIP + Branch-and-cut + Meta-heuristic
Rodriguez-Martin et al. (2016)	SA	Endogenous	Cost	$\leq q$ spokes per access ring + $1 \leq \text{access rings} \leq k$ per hub	MIP + Branch-and-cut
Gelareh et al. (2017a)	SA	$q = 3 \leq \dots \leq p$	Time (transit transshipment) +	Yes	MIP + Branch-and-cut + Benders decomposition
Rodriguez-Martin et al. (2014)	SA	Exogenous	Cost	$\leq q$ spokes per cycle	MIP + Branch-and-cut
Gelareh et al. (2013)	SA	Exogenous	Cost	$\geq q$ spokes per cycle	MIP + Lagrangian decomposition based heuristic
Çetiner et al. (2010)	MA	Endogenous	Cost + Num. of vehicles	No	Heuristic

TABLE I: A summary of the relevant contributions in the literature

From a theoretical and modeling perspective, this work extends the Hub Location Problem (HLP) by incorporating cycle-based redistribution and generalizing previous research on cost-efficient distribution networks. We introduce a formal mathematical representation of the problem, capturing hub placement, pharmacy assignment, and intra-cycle redistribution costs. However, due to the computational complexity of exact MIP solvers, we propose a non-exact hyper-heuristic solution approach to efficiently handle large-scale instances. The non-exact approach ensures practical feasibility in real-world applications.

The contributions of this work can be summarized as follows: We introduce a novel mathematical formulation that optimizes the total cost of hub setup and supply delivery while incorporating cycle-based redundancy. Our approach balances hub placement, optimal hub network and pharmacy-to-pharmacy redistribution, ensuring cost minimization without relying entirely on centralized hub connections. Additionally, we propose a heuristic-based solution method to efficiently solve the model, allowing it to handle large-scale instances where exact optimization techniques might be impractical. Finally, we conduct extensive computational experiments to validate the effectiveness and scalability of the proposed approaches, demonstrating significant cost reductions and improved network performance compared to traditional approaches.

The remainder of this paper is structured as follows: Section II provides a formal definition of the problem and

introduces the cycle-based distribution framework. Section III presents the MIP formulation, detailing the optimization structure and constraints. Section 4 describes the solution approach, including heuristic-based methods for efficiently solving the model. Section 5 reports the results of computational experiments, analyzing the performance of the proposed model across different scenarios. Finally, Section 6 concludes the paper with key insights, practical implications, and suggestions for future research directions.

## II. PROBLEM STATEMENT

The Cycle  $p$ -Hub Location Problem (CpHLP) (Figure 2) can be formally described as follows: Given a set of nodes  $V$  where  $|V| = n$ , a cost matrix  $C$  where  $c_{ij}$  is the cost per unit of flow on the edge  $i$  to  $j$ , a flow/demand matrix  $W$  where  $w_{ij}$  is the flow to be sent from  $i$  to,  $j$ . The fixed costs of setting up hub nodes, hub edges and spoke arcs are denoted by  $F_k$ ,  $I_{kl}$  and  $S_{kl}$ , respectively. The problem is to find  $p$  nodes called hub nodes and establish the hub-level network using hub edges, connecting hub nodes in such a way that the hub-level network remains connected with undirected connections (edges). The remaining  $n - p$  spoke nodes will be allocated to different hub nodes from among the  $p$  hubs. The spoke-level nodes form directed rotation services starting from the hub node, visiting all the spoke nodes allocated to it, and returning to the hub node. There will be at least two spoke nodes allocated to every hub node unless practical real-life situations do not

permit this, in which case the solely allocated spoke node is called an isolated spoke node and is spurred at the hub node. A discount factor,  $\alpha$ , represents the economies of scale at the hub-level network. The objective is to find the optimal structure, minimizing the total costs associated with setting up facilities and transportation costs for such a structure.

### III. MIXED-INTEGER LINEAR PROGRAMMING FORMULATIONS

The variables of model follow:  $x_{ij} = 1$  ( $\forall i \neq j$ ), if node  $i$  is allocated to hub  $j$ , 0, otherwise;  $h_i = 1$  if  $i$  is a hub,

0, otherwise;  $y_{ij} = 1$  ( $\forall i \neq j$ ), if there exists a spoke arc  $(i, j)$ , 0 otherwise;  $b_{ij} = 1$  ( $\forall i < j$ ), if a hub edge  $\{i, j\}$  is established between two hub nodes  $i$  and  $j$ , 0 otherwise;  $w_{ijkl}$  represents the fraction of flow from  $i$  to  $j$  routed via hub edge  $(k, l)$  and  $s_{ijkl}$  stands for the fraction of flow from  $i$  to  $j$  routed via spoke arc  $(k, l)$ .

The Cycle  $p$ -Hub Location Problem with Runaway (CpHLP) model can be stated as follows:

$$\min \sum_{i=1}^n \sum_{\substack{j=1 \\ j \neq i}}^n \sum_{\substack{k=1 \\ k \neq j}}^n \sum_{\substack{l=1 \\ l \neq k \\ l \neq i}}^n w_{ijkl} c_{kl} (s_{ijkl} + \alpha w_{ijkl}) + \sum_{k=1}^n \sum_{\substack{l=1 \\ l > k}}^n I_{kl} b_{kl} + \sum_{k=1}^n \sum_{\substack{l=1 \\ l \neq k}}^n S_{kl} y_{kl} + \sum_{k=1}^n F_k h_k \quad (1)$$

$$s.t. \sum_{j=1}^n h_j = p, \quad (2)$$

$$x_{ij} \leq h_j, \quad \forall i, j \in V, j \neq i, \quad (3)$$

$$\sum_{j=1, j \neq i}^n x_{ij} + h_i = 1, \quad \forall i \in V, \quad (4)$$

$$y_{ij} + x_{ik} \leq 1 + x_{jk}, \quad \forall i, j, k \in V, k \neq j, k \neq i, j \neq i, \quad (5)$$

$$y_{ij} + h_i \leq 1 + x_{ji}, \quad \forall i, j, k \in V, k \neq j, j \neq i, \quad (6)$$

$$y_{ij} + x_{ij} \leq 1 + h_j, \quad \forall i, j, k \in V, k \neq i, j \neq i, \quad (7)$$

$$y_{ij} + y_{ji} + x_{ki} + x_{kj} \leq 3 - h_i - h_j, \quad \forall i, j, k \in V, k \neq j, k \neq i, j > i, \quad (8)$$

$$\sum_{j=1, j \neq i}^n y_{ij} = 1, \quad \forall i \in V, \quad (9)$$

$$\sum_{j=1, j \neq i}^n y_{ji} = 1, \quad \forall i \in V, \quad (10)$$

$$b_{ij} \leq h_i, \quad \forall i, j \in V, j > i, \quad (11)$$

$$b_{ij} \leq h_j, \quad \forall i, j \in V, j > i, \quad (12)$$

$$\sum_{l=1, l \neq i}^n w_{ijil} + \sum_{l=1, l \neq i}^n s_{ijil} = 1, \quad \forall i, j \in V, j \neq i, \quad (13)$$

$$\sum_{l=1, l \neq j}^n w_{ijlj} + \sum_{l=1, l \neq j}^n s_{ijlj} = 1, \quad \forall i, j \in V, j \neq i, \quad (14)$$

$$\begin{aligned} & \sum_{\substack{l=1 \\ l \neq i \\ l \neq k}}^n w_{ijkl} + \sum_{\substack{l=1 \\ l \neq i \\ l \neq k}}^n s_{ijkl} \\ &= \sum_{\substack{l=1 \\ l \neq j \\ l \neq k}}^n w_{ijlk} + \sum_{\substack{l=1 \\ l \neq j \\ l \neq k}}^n s_{ijlk}, \end{aligned} \quad \forall i, j, k \in V, k \neq i, k \neq j, j \neq i, \quad (15)$$

$$s_{ijkl} \leq y_{kl}, \quad \forall i, j, k, l \in V, l \neq k, l \neq i, k \neq j, j \neq i, \quad (16)$$

$$w_{ijkl} + w_{ijlk} \leq b_{kl}, \quad \forall i, j, k, l \in V, l > k, j \neq i, \quad (17)$$

$$h_i, x_{ij}, y_{ij}, b_{ik} \in \{0, 1\}, \quad \forall i, j, k, l \in V, j \neq i, k > i, \quad (18)$$

$$w_{ijkl}, s_{ijkl} \in (0, 1),$$

$$\forall i, j, k, l \in V, l \neq k, l \neq i, k \neq j, j \neq i. \quad (19)$$

The objective function minimizes the transportation costs and the setup costs for hubs, hub edges, and spoke arcs. Constraints (2) ensure that the number of hubs is equal to  $p$ . Constraints (3) guarantee that a node  $i$  can be allocated to node  $j$ , only if node  $j$  is a hub. Constraints (4) ensure that a node  $i$  is either a hub node or is allocated to only one hub. Constraints (5), (6) and (7) guarantee that a spoke arc links two spokes of the same hub (cycle). Constraints (8) are aggregations of the two constraints  $y_{ij} + y_{ji} \leq 1$  and  $x_{ki} + x_{kj} \leq 2 - h_i - h_j$ . The former makes sure that two spoke arcs in opposite directions do not exist, and the latter guarantees that one spoke node cannot be allocated to two hub nodes at the same time.

Constraints (9) ensure that a spoke  $i$  has exactly one outgoing spoke arc. Constraints (10) ensure that a spoke  $i$  has exactly one incoming spoke arc. Constraints (11) and (12) ensure that if the hub edge  $b_{ij}$  exists, then the nodes  $i$  and  $j$  are hubs. Constraints (13) assure an origin-destination flow  $i - j$  leaves  $i$  via a spoke or a hub connection. Constraints (14) assure that an origin-destination flow  $i - j$  arrives at its destination  $j$  via a spoke or a hub connection. Constraints (15) assure that flow conservation holds at every intermediate node visited along an origin-destination path for flow  $i - j$ . Constraints (16) and (17) guarantee that a flow between two nodes  $i$  and  $j$  will traverse the link  $k - l$  (spoke arc or hub edge) if a compatible link exists. Constraints (18) are integrality constraints.

#### IV. METAHEURISTIC AND HYPER-HEURISTIC OPTIMIZATION APPROACH (PLANNED)

While this paper focuses on the model, future work will involve the design of:

- Metaheuristics (e.g., Genetic Algorithms) for hub selection and cycle optimization.
- Hyper-heuristics for dynamic heuristic selection.
- Comparative analysis with traditional models.

##### A. Why Meta-heuristics?

Exact methods are computationally expensive, making meta-heuristics a preferred choice for large-scale optimization.

#### V. CONCLUSION AND FUTURE WORK

This paper presents a novel cost optimization framework for pharmaceutical distribution using a cycle-based hub location model. We formulated the problem as a mixed-integer linear program, emphasizing theoretical modeling and network design. Although heuristic solution methods and computational experiments are planned for future work, the current formulation lays the foundation for scalable and flexible pharmaceutical supply networks.

#### ACKNOWLEDGMENT

This work was supported by the Directorate General of Scientific Research and Technological Development (DGRSDT), Ministry of Higher Education and Scientific Research, and the PGMO program.

#### REFERENCES

- Alumur, S. and Kara, B. Y. (2008). Network hub location problems: The state of the art. *European Journal of Operational Research*, 190(1):1–21.
- Alumur, S. A., Yaman, H., and Kara, B. Y. (2012). Hierarchical multimodal hub location problem with time-definite deliveries. *Transportation Research Part E: Logistics and Transportation Review*, 48(6):1107 – 1120.
- AlZaidan, M., Hadid, M., Padmanabhan, R., and Kerbache, L. (2024). Optimizing pharmaceutical supply chain configuration in primary healthcare: A mathematical modeling and decision support approach. In Bruzzone, A., Frascio, M., Longo, F., and Novak, V., editors, *13th International Workshop on Innovative Simulation for Health Care, IWISH 2024*, Proceedings of the International Workshop on Innovative Simulation for Health Care, IWISH, Italy. Cal-Tek srl.
- Azizi, N. (2019). Managing facility disruption in hub-and-spoke networks: formulations and efficient solution methods. *Annals of Operations Research*, 272(1):159–185.
- Benabbou, L., Berrado, A., and Haial, A. (2020). Redesigning a transportation network: the case of a pharmaceutical supply chain. *International Journal of Logistics Systems and Management*, 35:90.
- Campbell, J. F. (1992). Location and allocation for distribution systems with transshipments and transportation economies of scale. *Annals of Operations Research*, 40(1):77–99.
- Campbell, J. F. (1994). Integer programming formulations of discrete hub location problems. *European Journal of Operational Research*, 72(2):387 – 405.
- Cardoso, S. R., Barbosa-Póvoa, A. P., Relvas, S., and Novais, A. Q. (2015). Resilience metrics in the assessment of complex supply-chains performance operating under demand uncertainty. *Omega*, 56:53 – 73.
- Carello, G., Croce, F. D., Ghirardi, M., and Tadei, R. (2004). Solving the hub location problem in telecommunication network design: A local search approach. *Networks*, 44:94–105.
- Çetiner, S., Sepil, C., and Süral, H. (2010). Hubbing and routing in postal delivery systems. *Annals of Operations Research*, 181(1):109–124.
- Chaharsooghi, S., Momayezi, F., and Ghaffarinasab, N. (2016). An adaptive large neighborhood search heuristic for solving the reliable multiple allocation hub location problem under hub disruptions. *International*

- Journal of Industrial Engineering Computations*, 8:191–202.
- Contreras, I. and Fernández, E. (2012). General network design: A unified view of combined location and network design problems. *European Journal of Operational Research*, 219(3):680–697. Feature Clusters.
- Contreras, I., Tanash, M., and Vidyarthi, N. (2016). Exact and heuristic approaches for the cycle hub location problem. *Annals of Operations Research*.
- Danach, K., Gelareh, S., and Neamatian Monemi, R. (2019). The capacitated single-allocation p-hub location routing problem: a lagrangian relaxation and a hyper-heuristic approach. *EURO Journal on Transportation and Logistics*.
- de Sá, M., Contreras, I., and Cordeau, J.-F. (2015a). Exact and heuristic algorithms for the design of hub networks with multiple lines. *European Journal of Operational Research*, 246(1):186 – 198.
- de Sá, M., Elisangela, Contreras, I., Cordeau, J.-F., Saraiva de Camargo, R., and de Miranda, G. (2015b). The hub line location problem. *Transportation Science*, 49(3):500–518.
- Dongjiu, L., Hao, L., Qingnian, Z., and Jiali, W. (2012). Transport hub-and-spoke network optimization model construction of pharmaceuticals cold-chain logistics. In *2012 IEEE Ninth International Conference on e-Business Engineering*, pages 304–307.
- Duarte, I., Mota, B., Pinto-Varela, T., and Barbosa-Póvoa, A. P. (2022). Pharmaceutical industry supply chains: How to sustainably improve access to vaccines? *Chemical Engineering Research and Design*, 182:324–341.
- Farahani, R. Z., Hekmatfar, M., Arabani, A. B., and Nikbakhsh, E. (2013). Hub location problems: A review of models, classification, solution techniques, and applications. *Computers & Industrial Engineering*, 64(4):1096–1109.
- Garcia, D. J. and You, F. (2015). Supply chain design and optimization: Challenges and opportunities. *Computers & Chemical Engineering*, 81:153–170. Special Issue: Selected papers from the 8th International Symposium on the Foundations of Computer-Aided Process Design (FOCAPD 2014), July 13-17, 2014, Cle Elum, Washington, USA.
- Gelareh, S., Maculan, N., Mahey, P., and Monemi, R. N. (2013). Hub-and-spoke network design and fleet deployment for string planning of liner shipping. *Applied Mathematical Modelling*, 37(5):3307 – 3321.
- Gelareh, S., Neamatian Monemic, R., and Semet, F. (2017a). Capacitated bounded cardinality hub routing problem: Model and solution algorithm. Technical report, arXiv:1705.07985.
- Gelareh, S., Neamatian Monemic, R., and Semet, F. (2017b). Capacitated bounded cardinality hub routing problem: Model and solution algorithm.
- Haial, A., Berrado, A., and Benabbou, L. (2016). A framework for designing a transportation strategy: The case of a pharmaceuticals supply chain. In *2016 3rd International Conference on Logistics Operations Management (GOL)*, pages 1–6.
- Kemmar, O., Bouamrane, K., and Gelareh, S. (2021). Hub location problem in round-trip service applications. *RAIRO - Operations Research*, 55:S2831–S2858.
- Kim, H. and O’Kelly, M. (2009). Reliable p-hub location problems in telecommunication networks. *Geographical Analysis*, 41:283 – 306.
- Kochakkashani, F., Kayvanfar, V., and Baldacci, R. (2024). Innovative applications of unsupervised learning in uncertainty-aware pharmaceutical supply chain planning. *IEEE Access*, 12:107984–107999.
- Low, Y., Halim, I., Adhitya, A., Chew, W., and Sharratt, P. (2016). Systematic framework for design of environmentally sustainable pharmaceutical supply chain network. *Journal of Pharmaceutical Innovation*, 11.
- Lucchese, A., Marino, A., and Ranieri, L. (2020). Minimization of the logistic costs in healthcare supply chain: a hybrid model. *Procedia Manufacturing*, 42:76–83. International Conference on Industry 4.0 and Smart Manufacturing (ISM 2019).
- Mohammadi, M., Tavakkoli-Moghaddam, R., Siadat, A., and Rahimi, Y. (2016). A game-based meta-heuristic for a fuzzy bi-objective reliable hub location problem. *Engineering Applications of Artificial Intelligence*, 50:1 – 19.
- Nasrollahi, M. and Razmi, J. (2021). A mathematical model for designing an integrated pharmaceutical supply chain with maximum expected coverage under uncertainty. *Operational Research*, 21.
- Neamatian Monemi, R., Gelareh, S., Nagih, A., Maculan, N., and Danach, K. (2021). Multi-period hub location problem with serial demands: A case study of humanitarian aids distribution in lebanon. *Transportation Research Part E: Logistics and Transportation Review*, 149:102201.
- Nozari, H. and Rahmaty, M. (2023). Optimization of the hierarchical supply chain in the pharmaceutical industry. *Edelweiss Applied Science and Technology*, 7:104–123.
- O’Kelly, M. (1987). A quadratic integer program for the location of interacting hub facilities. *European Journal of Operational Research*, 32:393–404.
- O’Kelly, M. (1992). Hub facility location with fixed costs. *Papers in Regional Science*, 71:293–306.
- Potters, E., Mosalla Nezhad, B., Bernard, V. J., Hans, E., and Asadi, A. (2024). Enhancing pharmaceutical cold supply chain: Integrating medication synchronization and diverse delivery modes. *International Transactions in Operational Research*.
- Rodriguez-Martin, I., Salazar González, J. J., and Yaman, H. (2014). A branch-and-cut algorithm for the hub location and routing problem. *Computers & Operations Research*, 50:161–174.
- Rodriguez-Martin, I., Salazar González, J. J., and Yaman, H. (2016). The ring k-rings network design problem: Model and branch-and-cut algorithm. *Networks*,

- 68:130–140.
- Rostami, B., Kämmerling, N., Buchheim, C., and Clausen, U. (2018). Reliable single allocation hub location problem under hub breakdowns. *Computers & Operations Research*, 96:15 – 29.
- Ward, K. K. (2017). *A Framework for Centralizing Inventory in Pharmaceutical Supply Chains*. Phd thesis, Wright State University.
- Yahyaei, M., Bashiri, M., and Randall, M. (2019). A model for a reliable single allocation hub network design under massive disruption. *Applied Soft Computing*, 82:105561.
- Yaman, H., Y.Kara, B., and Ç.Tansel, B. (2007). The latest arrival hub location problem for cargo delivery systems with stopovers. *Transportation Research Part B: Methodological*, 41(8):906 – 919.
- Yang, K., Liu, Y., and Yang, G. (2013). An improved hybrid particle swarm optimization algorithm for fuzzy p-hub center problem. *Computers & Industrial Engineering*, 64(1):133 – 142.
- Yuan, Z. and Gao, J. (2022). Dynamic uncertainty study of multi-center location and route optimization for medicine logistics company. *Mathematics*, 10(6).
- Zhalechian, M., Torabi, S. A., and Mohammadi, M. (2018). Hub-and-spoke network design under operational and disruption risks. *Transportation Research Part E: Logistics and Transportation Review*, 109:20 – 43.
- Zhong, W., Juan, Z., Zong, F., and Su, H. (2018). Hierarchical hub location model and hybrid algorithm for integration of urban and rural public transport. *International Journal of Distributed Sensor Networks*, 14(4).

# A Semantic User Profile with Machine Learning for Social Network Communities

Samia Bousalem

*LIRE Laboratory*

*University of Constantine 2 - Abdelhamid Mehri*

Constantine, Algeria

samia.bousalem@univ-constantine2.dz

Massinissa Chelghoum

*LIRE Laboratory*

*University of Constantine 2 - Abdelhamid Mehri*

Constantine, Algeria

massinissa.chelghoum@univ-constantine2.dz

Habib-Ellah Guergour

*LIRE Laboratory*

*University of Constantine 2 - Abdelhamid Mehri*

Constantine, Algeria

habib.guergour@univ-constantine2.dz

Karim Zarour

*LIRE Laboratory*

*University of Constantine 2 - Abdelhamid Mehri*

Constantine, Algeria

karim.zarour@univ-constantine2.dz

**Abstract**—Social network communities have become integral to modern digital ecosystems, enabling user communication, collaboration, and content sharing. In this context, effectively representing and utilizing user profiles is essential for enhancing user experience and engagement. This article presents our contribution: a User Profile Ontology (UPO), designed to model user profiles using semantic technologies for personalized recommendations. Our approach structures user characteristics and integrates machine learning techniques to predict users' centers of interest. Specifically, we employ the Naïve Bayes (NB) algorithm, achieving an accuracy of 84% in classifying user interests based on their interactions and behaviors. This hybrid methodology enhances the precision of recommendations, ensuring that users receive content aligned with their evolving preferences. By leveraging semantic web technologies and machine learning, our model facilitates intelligent decision-making across web resources, ultimately optimizing user engagement and personalization in social network communities.

**Index Terms**—User Profile, Ontology, Machine Learning, Personalization, Social Goals, Naïve Bayes, Interest Prediction

## I. INTRODUCTION

Since their inception, recommendation and information personalization mechanisms have faced challenges in providing users with relevant resources tailored to their needs and expectations. Despite heavy reliance on the user profile concept, these systems predominantly use a syntactic approach to process profile elements, often yielding inadequate outcomes. Constructing a semantically rich user profile is crucial for enabling recommendation and information retrieval systems to accurately interpret and utilize user data. However, establishing a standardized vocabulary for modeling user profiles in social network contexts is challenging due to the diverse nature of social networks and the varied data users handle. Despite existing efforts in semantic modeling of user profiles, each study introduces its vocabulary [1] [2].

Furthermore, the iterative and continuous construction and enrichment of user profiles as users engage in social networks [3] [4] have led to a vast amount of knowledge emerging

from user interactions. This wealth of information makes evolving ontology more challenging, especially in integrating all crucial information about user activities into the user profile ontology. This complexity arises from the diverse uses of social networks and the heterogeneous data handled by users, making it essential to incorporate all significant details into the user profile ontology to ensure its effectiveness [5].

Additionally, the dynamic nature of user interactions and evolving social network platforms introduces further complexities in maintaining the relevance and accuracy of the user profile ontology. Changes in user behavior, platform features, and the emergence of new interaction types require continuous updates and revisions to the ontology. Balancing the need for comprehensive coverage of user activities with the agility to adapt to evolving trends and patterns poses a significant challenge in ontology evolution.

One of the major challenges in user profiling is the lack of explicit information provided by users, making it difficult to build comprehensive and meaningful profiles. Many users do not fully specify their interests, preferences, or personal attributes, limiting user profile models' effectiveness. It is essential to automatically infer users' centers of interest based on their interactions within social networks to address this issue. Machine learning techniques can be leveraged to analyze user activities, detect behavioral patterns, and predict relevant interests, thus enriching the semantic user profile. By integrating machine learning, we ensure that user profiles remain complete and up-to-date, even in cases where direct user input is limited.

To address these challenges, we propose a user profile based on semantic representation. This approach incorporates a universal vocabulary, ensuring consistent interpretation across different systems. Establishing a common semantic framework enables seamless communication and interoperability among diverse recommendation and information retrieval systems. Moreover, to enhance the adaptability and effectiveness of

the semantic user profile, we propose integrating machine learning techniques to automatically enrich user profiles with inferred interests. Through continuous learning and adaptation, the semantic user profile remains dynamic and responsive to evolving user preferences and behaviors.

Besides this introduction, the remainder of the paper is structured as follows. Section 2 summarizes relevant approaches for user profiling in social network communities. Section 3 describes the proposed approach for user modeling, detailing the semantic user profile construction and the machine learning-based user interest prediction. Finally, the conclusion to our work is given in Section 4.

## II. RELATED WORK

Various approaches have been explored to enhance the representation and utilization of user data in the domain of user profile modeling using ontologies. Existing research and applications have focused on developing specialized ontologies tailored to user profiles.

One notable study, [6] has concentrated on constructing a user profile ontology that incorporates essential concepts and properties for structuring user profiles. The objective is to establish a comprehensive reference point within ontological frameworks for organizing user profiles effectively. Another significant contribution, [7] addresses the necessity of ontologies in user profile creation by exploring the development of ontologies specifically designed for constructing user profiles. This work emphasizes the importance of robust ontological structures in capturing diverse user data efficiently. Additionally, [8] defines the user profile and presents a methodology for ontological modeling of user profiles, highlighting the crucial role of ontologies in enhancing E-orientation platforms.

Another existing approach in user profile modeling using ontologies involves dynamic user profile ontology construction. This method emphasizes the dynamic construction of user profile ontologies that evolve in real time, responding to user interactions and behavior. Integrating dynamic elements into the ontology construction process aims to continuously adapt user profiles to reflect the latest user preferences and activities. The focus on real-time adaptation ensures that user profiles remain relevant and up-to-date, aligning closely with users' evolving needs and behaviors. Research by [9] explores the influence of online personalization and its trajectory through time. It incorporates insights from human-computer interaction research, specifically emphasizing Media Equation Theory, which posits that computers are perceived as social entities.

These studies collectively underscore the significance of leveraging ontologies to structure user information. Thus, ontologies enhance personalized experiences and provide valuable insights into the development and application of ontologies to improve user profiling methodologies.

## III. BACKGROUND CONCEPTS

### A. User Profile

A user profile refers to a collection of attributes, characteristics, and preferences associated with an individual user within

a specific context, such as a social network community. It represents the user's identity, interests, activities, and interactions within the digital environment. User profiles are crucial in facilitating personalized experiences, content recommendations, and targeted interactions within online platforms.

### B. Ontology

In information science and knowledge representation, an ontology is a formal and explicit specification of a conceptualization [10]. It defines a set of concepts, properties, and relationships within a specific domain of interest. Ontologies structure and organize knowledge systematically, enabling effective information retrieval, reasoning, and knowledge sharing across different systems and applications.

### C. Existing Ontologies

Existing ontologies refer to pre-existing conceptual frameworks and knowledge structures developed within various domains. These ontologies are designed to capture and represent domain-specific knowledge, including concepts, entities, relationships, and constraints relevant to a particular field of study or application. Table 1 summarizes each existing ontology.

TABLE I  
SUMMARY FOR EACH EXISTING ONTOLOGY

Existing ontologies	Definition
FOAF	defines a vocabulary for describing people and their relationships in a social network context.
SIOC	designed to represent information from online communities, forums, and discussion boards in a machine-readable format.
OPO	models online presence and availability information of individuals across different online platforms and social networks.
SWUM	focuses on modeling user profiles and preferences in the context of the Semantic Web.
SemSNI	designed to integrate and represent data from multiple social networking sites and platforms in a unified semantic format.

## IV. APPROACH FOR USER MODELING

In this section, we provide an overview of the proposed semantic user profile, which is the foundation for our approach to user modeling in social network communities. The semantic user profile is designed to capture and represent user preferences, interests, and interactions in a structured and semantic format. Unlike traditional user profiles that rely on syntactic processing, the semantic user profile leverages semantic technologies to enhance the richness and flexibility of user representation.

To achieve our objectives, our approach consists of three main phases, as shown in Figure 1:

- Semantic modeling of the user profile;
- Detection of changes by analyzing user traces in their social network;
- Application of changes.

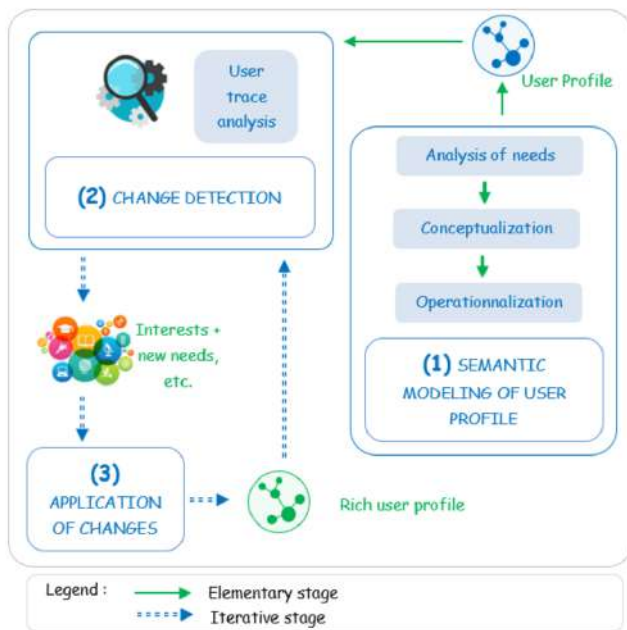


Fig. 1. A Semantic-Based Approach for Modeling and Enriching User Profiles

The first phase (1) involves constructing a semantic representation of the user profile, starting with the proposal of a profile ontology building process. This phase lays the foundation for the subsequent phases by establishing a structured framework for capturing user preferences and interests. In the second phase (2), our focus shifts to deducing new knowledge based on the initial profile obtained in the previous phase. This phase takes as input a set of user activities collected from social networks and the user's needs selected from the user profile ontology. By analyzing these inputs, we aim to identify and extract additional elements of the user profile, particularly focusing on interests. The outcome of this phase is a set of new elements that contribute to enriching the user profile with relevant knowledge. Finally, the last phase (3) involves iteratively enriching the user profile with the results obtained in the previous phase. This iterative process ensures that the user profile remains dynamic and responsive to evolving user preferences and behaviors. By continuously updating and refining the user profile, we aim to maintain its relevance and effectiveness in supporting personalized recommendation and information retrieval services.

#### A. Ontology-Based User Profile Modeling

The first phase of our proposed approach utilizes semantic web technologies, specifically ontologies, along with the associated models, formalisms, languages, and tools derived from their application. However, to ensure the robustness and consistency of the user profile ontology, we propose a structured building process that provides clear guidance and organization for its development. To achieve this goal, we propose a process consisting of three tasks:

- **Needs Analysis:** The primary objective of the needs analysis is to delineate the purpose of building the ontology for the user profile. This involves finely delineating the aspects of the user profile, identifying relevant information sources, and examining existing ontologies, particularly those prevalent on the semantic and social web. By drawing inspiration from these sources, we aim to identify the pertinent terms and concepts for inclusion in the user profile ontology.

- **Conceptualization:** Following the needs analysis, the conceptualization phase involves analyzing the identified set of terms and organizing them into intermediate representations. This process entails coherently structuring the terms to facilitate their implementation in a formal and operational language. By establishing clear conceptual frameworks, we lay the groundwork for developing a robust and comprehensive user profile ontology.

- **Operationalization:** Once the terms are assembled and structured, the next step is to select an operational language for the user profile ontology that machines can process. This involves choosing a suitable formal language, such as OWL (Web Ontology Language), and encoding the ontology in a machine-readable format. During the operationalization phase, the user profile ontology is made so machines can process it. This makes it easier to use in semantic web environments and other systems.

In the following subsections, we delve into developing each of its tasks as follows:

1) *Needs Analysis:* This phase begins with defining the purpose for which the ontology will be built. As previously presented, the primary objective behind the development of a user profile ontology is to provide recommendation or information retrieval systems with an efficient and effective means to interpret the user profile, ensuring that the results provided by these systems best meet the needs of their users.

Once the goal is clearly defined, we collect the terms for the profile ontology. This task would be impossible without specifying the dimensions the user profile ontology must cover. Therefore, each dimension subsequently consists of a set of concepts and relationships, with dependencies on one another.

Several dimensions in the literature can cover the elements constituting the user profile [11] [12] [13]. However, in our approach, we define six dimensions adapted to the social web context. To this end, we carefully select vocabularies adapted to each dimension, starting by searching for those most widely adopted on the social web, such as FOAF (Friend Of A Friend) [14], SIOC (Semantically Interlinked Online Communities) [15], etc. Then, we consider those most used in online communities, namely OPO (Online Presence Ontology) [16], SWUM (Social Web User Model) [17], SemSNI (SEMantically Social Network Interactions) [18], etc. Finally, if none of the mentioned vocabularies adequately represent a dimension, we propose our own. Once the different parts of the vocabulary are acquired, we meticulously define relationships to link them together coherently. Below are the details for each dimension:

• **Activities and personal information:** In this dimension, all information about the user, specifically their personal details provided during initial registration on a social network, as well as their roles within the community, will be grouped.

• **Online Presence:** The "Online Presence" dimension encompasses details about the user's occupation and availability on social networks. This information is crucial for recommendation systems to suggest relevant resources at the right time, optimizing system functionality. User engagement duration within the social network contributes to deducing their online presence and enhancing the effectiveness of recommendation systems based on timing considerations.

• **Personality:** This dimension refers to the psychological characteristics, traits, and behaviors that uniquely define an individual's identity and influence their interactions and preferences within a social network or online community.

• **Needs:** A user's needs are typically reflected upon registration or when a social network explicitly requests the user's opinion and needs within an online community.

• **Interactions:** Within social networks, users can interact in their environment by communicating with other members, sharing resources, and expressing opinions about certain resources, among other actions. All of these actions will be grouped in the "Interactions" dimension.

• **Interests:** The "Interest" dimension encompasses the topics, subjects, or areas of focus that attract a user's attention and engagement within a social network or online community.

Table 2 provides a summary of the vocabulary used for each dimension.

TABLE II  
VOCABULARIES USED FOR EACH DIMENSION

Dimension	Used Vocabularies
Activities & personal information	SIOC, FOAF
Online Presence	OPO
Personality	/
Needs	SWUM
Interactions	SemSNI
Interest	/

2) *Conceptualization:* Once the dimensions are defined, the next step is to analyze each term set within the dimensions, select the appropriate terms, organize them, combine and relate them together, and then structure them into intermediate representations. These representations will be used later for their implementation in a formal and operational language. Table 3 lists all the terms of the profile ontology, particularly demonstrating how the concepts from different dimensions are interconnected to ensure consistency across all terms.

To complete the conceptualization, we have chosen the description logic formalism, specifically the SHIQ family [19], which corresponds to the OWL language. OWL will be used to codify the user profile ontology, providing a standardized and interoperable format for representation.

In what follows, we present formalizations of terms in the two parts comprising the SHIQ family: the Terminological part (TBOX) and the Assertional part (ABOX) (see Fig. 2).

TABLE III  
OVERVIEW OF RELATIONSHIPS AND CONCEPTS ACROSS DIMENSIONS

Relation	Domain	Range
hold_account	Foaf : Person	Sioc : UserAccount
Knows	Foaf : Person	Foaf : Person
Follows	Sioc : UserAccount	Sioc : UserAccount
has_creator	Sioc : Item	Sioc : UserAccount
DeclaredOn	Opo : OnlinePresence	Sioc : UserAccount
CurrentAction	Opo : OnlinePresence	Opo : Action
Has_personality	Sioc : UserAccount	UPO : Personality
hasNeeds	Foaf : Person	Swum : UserNeeds
Has_visitor	SemSNI : Visit	Sioc : UserAccount
Has_interest	Foaf : Person	UPO : Interest

```

PrivateMessage ⊆ ⊂ Comment ⊂ ⊂ Visit ⊂ (≥1 Recipient.UserAccount) ⊂ (≥1 Sender.UserAccount)

UserAccount ⊆ (∃ username.String) ⊂ (∃ password.String) ⊂ (≥1 Has_function.Role) ⊂ (≤3 Has_Personality.Personality)
... // and many other definition of concepts in TBOX
Has_interest (Person, Interest)
Has_Personality (UserAccount, Personality)
... // and many other definition of relations in TBOX
SAMIA: Person
MASSINIISA, Person
LEARNING_SAMIA: Learning
(SAMIA, MASSINIISA): Knows
(SAMIA, LEARNING_SAMIA): Has_Needs
..... and many other definitions of instances in ABOX

```

Fig. 2. Formalization of the TBOX and ABOX parts.

3) *Operationalization:* This step aims to formalize and implement the previously acquired terms, ensuring that the user profile ontology is operational. For the implementation process, we used the Protege tool [20] to automatically generate the OWL ontology code. To accomplish this, we implemented each dimension in a separate source file, and then integrated them into the overall ontology. Each ontology developed was associated with a namespace to maintain clarity and organization within the system. The figures presented below, Figure 3 and Figure 4, provide screenshots showcasing the implementation of the user profile ontology.

Tests have been meticulously conducted on the ontology model to assess its validity and consistency. The Hermit reasoner [21] was employed as a powerful tool to perform these evaluations. The reasoner systematically analyzed the ontology, checking for logical coherence and adherence to specified constraints. Through rigorous testing procedures, the ontology was verified to be both valid and consistent, ensuring its reliability and effectiveness in representing user profiles within social network communities (see Fig. 5).

### B. Predicting User Interests with Machine Learning

To enhance the user profile with inferred interests, we integrate machine learning techniques that automatically classify user preferences based on textual data. The objective of this step is to predict a user's center of interest by analyzing their textual interactions, such as posts, comments, or articles. By leveraging machine learning, we ensure a dynamic and continuously evolving user profile that adapts to changing

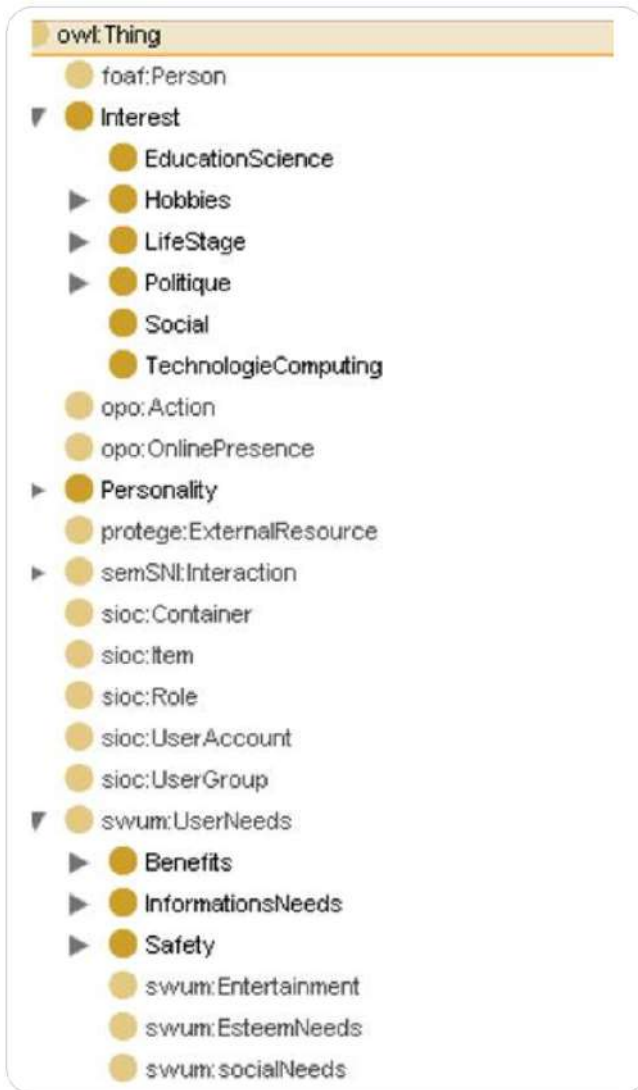


Fig. 3. User profile ontology class hierarchy.

preferences and behaviors. To effectively infer user interests, we employ a machine learning-based approach. This step consists of three key phases: data collection, where we acquire a relevant dataset for training, data preprocessing, where we clean and normalize the text data, and model training, where we train a classifier to predict user interests.

1) *Data Collection*: To train and evaluate our model for user interest prediction, we utilized the 20 Newsgroups dataset [22]. This publicly available dataset contains approximately 20,000 newsgroup documents categorized into 20 topics, covering domains such as politics, sports, technology, and science. The dataset is widely used for text classification tasks due to its rich diversity of textual data and well-structured categories. Each document in the dataset represents a real-world discussion thread from online newsgroups, making it highly suitable for training machine learning models to classify

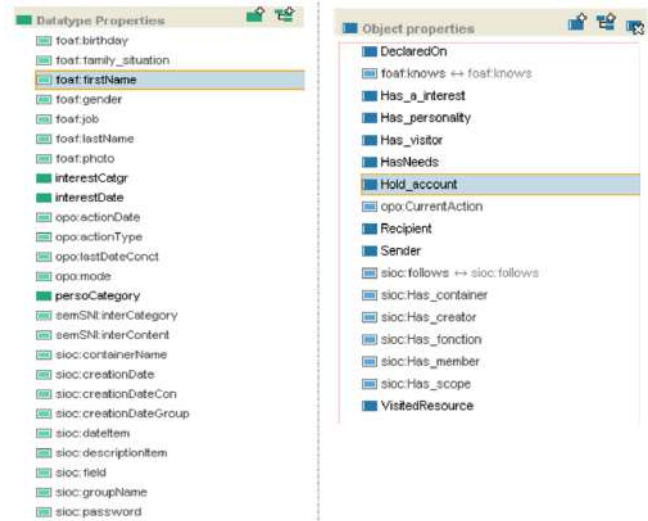


Fig. 4. Attributes and class relationships of the user profile ontology.

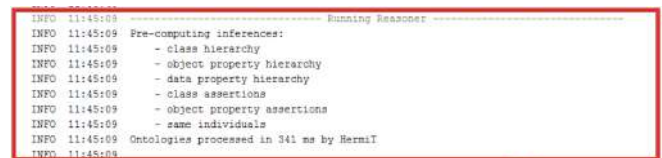


Fig. 5. A test of the learner profile ontology.

user interests based on textual content.

The 20 Newsgroups dataset is divided into two subsets: a training set and a test set. The training set builds the classification model by learning patterns from labeled textual data, while the test set evaluates the model's performance on unseen data.

2) *Data Preprocessing*: Before training the model, we apply preprocessing steps to clean and normalize the text data, ensuring better model performance. The main steps are:

- **Tokenization** : the text is split into individual words (tokens) based on delimiters such as spaces and punctuation marks. This allows the model to analyze text at the word level.
- **Stop Word Removal** : common words that do not add meaningful information (e.g., pronouns, prepositions, and articles such as the, is, and, etc.) are removed. This reduces noise in the dataset and focuses on relevant words.
- **Normalization** : this process ensures that different variations of the same word are treated as a single entity. We apply stemming (reducing words to their root form) or lemmatization (converting words to their dictionary base form).

3) *Model Training*: We employed the Naïve Bayes (NB) algorithm for the classification task, a probabilistic machine learning model commonly used for text classification due to its

efficiency and strong performance on sparse data. Naïve Bayes assumes conditional independence between words, making it computationally efficient and effective for high-dimensional datasets such as 20 Newsgroups.

- **Feature Extraction:** we transformed the text data into numerical vectors using TF-IDF representation.
- **Training the Model:** we trained a Multinomial Naïve Bayes classifier using the training set of the 20 Newsgroups dataset. This algorithm is well-suited for text classification as it calculates the probability of a document belonging to a specific category based on word frequencies. The training phase involves learning the statistical distribution of words across different interest categories.
- **Evaluation :** To assess the model's performance, we used the test set of the 20 Newsgroups dataset. The trained model was applied to unseen data, and its predictions were compared to the actual categories to measure accuracy. The classifier achieved 84% accuracy, demonstrating its effectiveness in predicting user interests from textual data. Figure 6 presents the classification report of the NB classifier, providing key performance metrics such as precision, recall, and F1-score for each interest category.

	NB	precision	recall	f1-score	support
alt.atheism	0	0.82	0.82	0.82	319
comp.graphics	1	0.69	0.74	0.71	389
comp.os.ms-windows.misc	2	0.73	0.62	0.67	394
comp.sys.ibm.pc.hardware	3	0.64	0.76	0.69	392
comp.sys.mac.hardware	4	0.82	0.82	0.82	385
comp.windows.x	5	0.83	0.79	0.81	395
misc.forsale	6	0.86	0.83	0.84	390
rec.autos	7	0.88	0.89	0.89	396
rec.motorcycles	8	0.95	0.94	0.95	398
rec.sport.baseball	9	0.96	0.92	0.94	397
rec.sport.hockey	10	0.95	0.97	0.96	399
sci.crypt	11	0.86	0.92	0.89	396
sci.electronics	12	0.79	0.74	0.77	393
sci.med	13	0.88	0.83	0.85	396
sci.space	14	0.89	0.89	0.89	394
soc.religion.christian	15	0.85	0.95	0.90	398
talk.politics.guns	16	0.81	0.91	0.86	364
talk.politics.mideast	17	0.97	0.93	0.95	376
talk.politics.misc	18	0.78	0.70	0.74	310
talk.religion.misc	19	0.75	0.65	0.70	251
accuracy				<b>0.84</b>	7532

Fig. 6. Classification report of the NB model for user interest prediction.

The high accuracy indicates that the NB approach is well-suited for interest prediction, as it effectively captures patterns in textual content and assigns users to relevant interest categories. To further validate the model's performance, we analyze the confusion matrix, which provides detailed insights into the classification results. The confusion matrix highlights the number of correctly and incorrectly classified instances across different interest categories, helping us assess the model's strengths and areas for improvement (see figure 7).

4) *Case Study: Predicting Donald Trump's Interests:* To validate our model on real-world data, we conducted a case

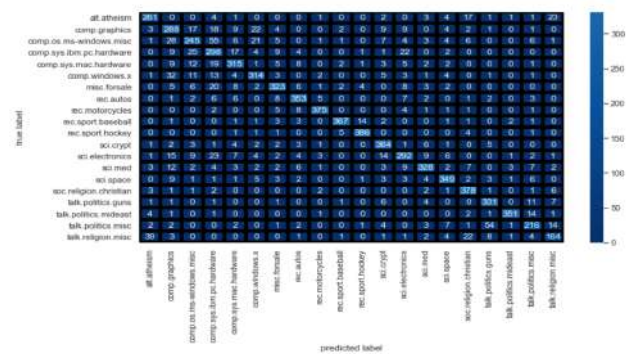


Fig. 7. Confusion matrix of the NB model for user interest prediction.

study analyzing Donald Trump's social media activities. By collecting and preprocessing a dataset of his tweets and public statements, we applied our trained NB classifier to determine his primary center of interest. We collected tweets and speeches related to Donald Trump from publicly available sources [23]. The text was then preprocessed through cleaning, tokenization, and vectorization using TF-IDF. Finally, the trained Naïve Bayes classifier was applied to the processed text, predicting his primary center of interest as "Politics".

This case study demonstrates the practical application of our user interest prediction model in analyzing real-world textual data, showcasing its ability to classify user interests based on linguistic patterns.

## V. CONCLUSION AND FUTURE WORKS

This paper presents a machine learning-based approach for constructing and enriching user profiles within social network communities. Our methodology integrates a semantic user profile model with machine learning techniques to infer user interests dynamically. We utilized the 20 Newsgroups dataset to train a Naïve Bayes classifier, demonstrating its effectiveness in accurately predicting user interests based on textual content. Additionally, we conducted a case study analyzing Donald Trump's activities, where our model successfully identified politics as his primary center of interest. To further validate the effectiveness of our approach, we examined the classification performance using evaluation metrics, including accuracy, the classification report, and the confusion matrix. The results confirmed that our approach effectively captures textual patterns and categorizes users into relevant interest groups. These findings highlight the potential of integrating semantic modeling with machine learning for user profiling.

As a next step, we propose extending this research to recommendation and information retrieval systems within social network communities. By leveraging our enriched user profiles, we aim to develop personalized recommendation systems capable of suggesting relevant resources tailored to users' inferred interests, enhancing user engagement and experience.

## REFERENCES

- [1] S. A. Tabrizi, A. Shakery, M. A. Tavallaei, and M. Asadpour, "Search personalization based on social-network-based interestedness measures," *IEEE Access*, vol. 7, pp. 119 332–119 349, 2019.
- [2] C.-J. Chu, I.-P. Chiang, K.-H. Tsai, and Y.-H. Tung, "Exploring the effects of personalized advertising on social network sites," *Journal of Social Media Marketing*, vol. 1, no. 2, pp. 38–54, 2022.
- [3] G. Vasanthakumar, K. Sunithamma, P. D. Shenoy, and K. Venugopal, "An overview on user profiling in online social networks," *Int. J. Appl. Inf. Syst.*, vol. 11, no. 8, pp. 25–42, 2017.
- [4] V. Oliseenko and M. Abramov, "Identification of user profiles in online social networks: a combined approach with face recognition," in *Journal of Physics: Conference Series*, vol. 1864, no. 1. IOP Publishing, 2021, p. 012119.
- [5] F. Azzam, M. Kayed, and A. Ali, "A model for generating a user dynamic profile on social media," *Journal of King Saud University-Computer and Information Sciences*, vol. 34, no. 10, pp. 9132–9145, 2022.
- [6] G. Maria, K. Akrivi, V. Costas, L. George, and H. Constantin, "Creating an ontology for the user profile: Method and applications," in *Proceedings AI\* AI Workshop RCIS*, 2007, pp. 407–412.
- [7] A. Barisic and M. Winckler, "Towards user profile meta-ontology," 2023.
- [8] J. Adib, R. A. Abdelouahid, A. Marzak, and H. Moutachauik, "Ontological user profile for e-orientation platforms," *Procedia computer science*, vol. 198, pp. 417–422, 2022.
- [9] M. Zanker, L. Rook, and D. Jannach, "Measuring the impact of online personalisation: Past, present and future," *International Journal of Human-Computer Studies*, vol. 131, pp. 160–168, 2019.
- [10] T. R. Gruber, "A translation approach to portable ontology specifications," *Knowledge acquisition*, vol. 5, no. 2, pp. 199–220, 1993.
- [11] M. Fernandez, A. Scharl, K. Bontcheva, and H. Alani, "User profile modelling in online communities," 2014.
- [12] T. Plumbaum, *User modeling in the social semantic web*. Technische Universitaet Berlin (Germany), 2015.
- [13] S. Ouafthouh, A. Zellou, and A. Idri, "User profile model: A user dimension based classification," in *2015 10th international conference on intelligent systems: Theories and applications (sita)*. IEEE, 2015, pp. 1–5.
- [14] D. Brickley and L. Miller, "Foaf vocabulary specification 0.91," 2007.
- [15] A. Passant, U. Bojars, J. G. Breslin, and S. Decker, "The sioc project: semantically-interlinked online communities, from humans to machines," in *International Workshop on Coordination, Organizations, Institutions, and Norms in Agent Systems*. Springer, 2009, pp. 179–194.
- [16] M. Stankovic, "Modeling online presence," in *J. Breslin, U. Bojars, A. Passant and S. Fernandez, editors, Proceedings of the First Social Data on the Web Workshop, Karlsruhe, Germany*, vol. 405. Citeseer, 2008, p. 58.
- [17] T. Plumbaum, S. Wu, E. W. De Luca, and S. Albayrak, "User modeling for the social semantic web," in *SPIM*, 2011, pp. 78–89.
- [18] G. Erétéo, M. Buffa, F. Gandon, and O. Corby, "Analysis of a real online social network using semantic web frameworks," in *The Semantic Web-ISWC 2009: 8th International Semantic Web Conference, ISWC 2009, Chantilly, VA, USA, October 25-29, 2009. Proceedings 8*. Springer, 2009, pp. 180–195.
- [19] B. Glimm, C. Lutz, I. Horrocks, and U. Sattler, "Conjunctive query answering for the description logic shiq," *Journal of Artificial Intelligence Research*, vol. 31, pp. 157–204, 2008.
- [20] M. A. Musen, *Automated generation of model-based knowledge acquisition tools*. Morgan Kaufmann Publishers Inc., 1989.
- [21] R. D. Shearer, B. Motik, and I. Horrocks, "Hermit: A highly-efficient owl reasoner," in *Owled*, vol. 432, 2008, p. 91.
- [22] K. Lang, "20 newsgroups dataset," 1995, accessed: 2025-03-24. [Online]. Available: <http://qwone.com/jason/20NewsGroups/>
- [23] A. Reese, "Trump tweets dataset," 2021, accessed: 2025-03-24. [Online]. Available: <https://www.kaggle.com/datasets/austinreese/trump-tweets>

# A Study of Aggregation Strategies for Ensemble Multi-Label Classifiers

Sonia Guehria<sup>1</sup>  
 Department of Computer Science  
 LRI Laboratory  
 Badji Mokhtar University  
 Annaba, Algeria  
[sonia.guehria@gmail.com](mailto:sonia.guehria@gmail.com)

Farida Kherissi<sup>2</sup>  
 Department of Financial Science  
 LISCO Laboratory  
 Badji Mokhtar University  
 Annaba, Algeria  
[kherissi.farida@gmail.com](mailto:kherissi.farida@gmail.com)

**Abstract**— Multi-label classification (MLC) is a challenging task in machine learning, where each instance can be associated with multiple labels simultaneously. A critical aspect of MLC is the aggregation of predictions in an ensemble, which plays a key role in improving classification performance by effectively combining outputs from multiple sub-models or leveraging label dependencies. Despite the critical role of aggregation methods in ensemble MLC, comprehensive comparative analyses and systematic investigations of these techniques remain notably lacking in the literature. This paper addresses this gap by proposing a comparative study of aggregation strategies for MLC and empirically evaluating their effectiveness across various benchmark datasets using multiple evaluation metrics. Our results demonstrate that the choice of aggregation strategy significantly impacts classification performance, with the weighted stacking strategy outperforming other approaches in most scenarios. This study not only advances the understanding of the aggregation process in MLC but also offers actionable recommendations for selecting the most suitable strategy for specific use cases.

**Keywords**— aggregation methods, ensemble approach, multi-label classification, label dependencies, data imbalance.

## I. INTRODUCTION

Multi-label classification (MLC) paradigm is widely encountered in numerous real-world applications and has attracted considerable attention from the machine learning and Data Mining communities over the past decades. Unlike traditional single-label classification, where each instance is assigned to only one class (either in binary or multi-class), MLC allows multiple labels to be associated with a single instance simultaneously. Formally, let  $X$  denote a set of instances and  $Y = \{1, 2, \dots, n\}$  a set of labels. Given a training sample  $S = \{(x_1, y_1), \dots, (x_m, y_m)\}$ , where  $x_i \in X$  is an instance and  $y_i \subseteq Y$  is the labelset associated with  $x_i$ , the goal is to design a multi-label classifier  $H$ , capable of predicting a set of labels for a new instance [1].

MLC is widely used in applications such as text categorization [2], [3], where a document may belong to multiple topics (e.g., "sports," "technology," and "politics"), and functional genomics [4], [5], where a gene can be associated with multiple functions (e.g., metabolism, protein synthesis, and transcription). Other domains include multimedia analysis [6], [7], [8], [9], [10], tag recommendation [11], web and rule mining [12], [13] map labeling [14], food trucks recommendation [15], and human activity recognition in smart homes [16]. Despite its

widespread use, MLC presents several challenges, including *label dependencies*, *high-dimensionality of the output spaces*, and *Imbalanced labels* [17]. To address these issues, *Ensemble* methods have been developed on top of the problem transformation or algorithm adaptation methods [18]. However, the effectiveness of ensemble-based MLC models depends significantly on aggregation strategies, which play a pivotal role in improving classification performance.

The main role of aggregation strategies is to combine predictions from multiple sub-models to enhance the robustness and generalization of MLC systems by leveraging two key principles: i) *Complementarity*: Sub-models compensate for each other's weaknesses while capitalizing on their strengths [19]. For example, some classifiers may perform well on rare labels, while others capture label correlations more effectively. Aggregating their outputs leads to more balanced and accurate predictions; ii) *Diversity*: Sub-models should exhibit varied behaviors, achieved by using different base algorithms, training on diverse data subsets, or varying feature sets [20]. If all models make similar errors, the ensemble's performance will not improve significantly. Despite the wide range of aggregation strategies available—from simple to advanced methods—their underlying mechanisms remain poorly explored in the field of Ensemble Multi-Label Classification (EMLC). As evidenced in the literature, only a limited number of studies have proposed dedicated combiner techniques for MLC [21], [22], [23].

To the best of our knowledge, this paper is the first to comprehensively address aggregation strategies for building ensemble models in MLC. We present a thorough comparative study of these strategies and conduct an empirical evaluation of their effectiveness. We categorize aggregation techniques into simple and advanced strategies and assess their performance across various benchmark datasets using different evaluation metrics. The rest of this paper is organized as follows. Section 2 provides background information, reviewing the state-of-the-art EMLC approaches used in our study and discussing the most popular aggregation strategies employed in ensemble models for MLC. Section 3 describes the operating principle of these aggregation methods. In Section 4, the experimental design, including datasets and evaluation metrics, is presented. The results of the comparative analysis of aggregation techniques are detailed and discussed in Section 5. Finally, Section 6 concludes the study and outlines future research directions.

## II. BACKGROUND

In this section, we formally define the ensemble approach for MLC and provide a concise overview of the advanced ensemble models examined in this study. Additionally, we discuss the most popular aggregation strategies employed in ensemble models for MLC.

### A. Formal Definition of EMLC

In the general context of MLC models, each classifier is responsible for predicting the presence or absence of a specific label. The predictions from all classifiers are then combined to determine the final subset of labels for a new instance. However, in ensemble models for MLC, an initial pool is generated by  $N$  multi-label classifiers  $C_1, C_2, \dots, C_N$ . For a new instance  $x_i \in X$  (the set of instances), each  $C_i$  generates a  $D$ -dimensional probability vector  $P_k$  such that  $P_k = [P_{1k}, P_{2k}, \dots, P_{mk}]$ . The probability assigned to each label  $L_i$  by an individual classifier  $C_i$  is then aggregated using a specific strategy to produce the final multi-label prediction [24].

### B. Ensemble Learning in MLC

Ensemble methods, whether based on *Problem Transformation* or *Algorithm Adaptation* approaches [18], aim to overcome the limitations of individual multi-label classifiers by combining multiple sub-models. In our study, we employ a complementary EMLCs, where each method addresses specific challenges in MLC.

ECC (Ensemble of Classifier Chains) [25] trains multiple CC classifiers over a random subset of instances and a randomly ordered chain of binary classifiers. In each chain, the feature space of a classifier is augmented with the predictions of preceding classifiers, explicitly capturing *label dependencies*. The final multi-label prediction is obtained by averaging probabilities across all base classifiers and applying a thresholding function to select the most relevant labels.

RAkEL (RANdom k-labelELsets) [26] constructs multiple LPs (Label Powerset) classifiers by randomly partitioning the label space into  $N$  subsets of size  $k$  (k-labelsets). Each LP predicts binary prediction for its assigned k-labelset, and the final multi-label prediction is generated by majority voting per label. RAKEL tackles *label imbalance* by ensuring rare labels are frequently grouped with frequent ones in small k-labelsets, preventing exclusion during training.

RF-PCT (Random Forest of Predictive Clustering Trees) [27] combines multiple PCTs (Predictive Clustering Trees) [28], each trained on random subsets of instances and features. The final multi-label prediction is obtained by aggregating the predictions from all PCTs through probabilistic voting. RF-PCT mitigates the *high dimensionality* through its hierarchical structure.

### C. Ensemble Technique

The construction of EMLCs incorporates either simple or advanced strategies. Our study focuses on *Majority Voting* as a simple strategy and on *Stacking* and *Weighted Stacking* as advanced strategies, with a weighting mechanism applied to specific labels.

*Majority Voting* is a foundational aggregation strategy for building an EMLC, it is used to aggregate predictions from individual classifiers for each label. Most EMLCs in the literature rely on majority voting [26], [29], [30], which helps reduce noise and uncertainty in predictions. However, this

approach ignores label dependencies and performs sub-optimally when the number of classifiers is limited [31]. Moreover, it treats all labels equally important- an assumption that often contradicts real-world scenarios where label relevance varies.

Recently, significant attention has been given to advances in Stacking for Multi-Label Learning. This approach has shown particularly promising results compared to other ensemble techniques, leading to its diverse applications. For instance, it is employed to aid in feature selection for identifying label-specific subsets [32], [33]. Similarly, it has been used to merge predictions from base classifiers using different rules [34], [35]. In other studies, this approach has been employed in hierarchical structures to reduce sampling complexity [144] and in pruning methods to refine the most relevant labels, preserving useful information for accurate predictions [36], [37]. Stacking has proven effective by leveraging historical sample data and allowing sub-models to adjust predictions based on corrected errors. However, it overlooks the effect of pairwise label dependencies [38].

To mitigate the limitations of the ensemble techniques mentioned earlier, the weighted approach has been applied in various real-world MLC applications [39], [40]. This allows the model to adapt more precisely to the specific characteristics and requirements of the problem. Consequently, for building more accurate systems, the Weighted approach has proven particularly well-suited for the Stacking framework [41], [42], [43], [44]. The estimation of diverse weights at various levels increased the accuracy of the Stacking process by generating more relevant predictions. Some studies have focused on weighting base classifiers [41], [45], [42], while others have investigated feature weighting [152], and many have examined the impact of label weighting [1].

## III. AGGREGATION STRATEGIES FOR MLC

### A. Majority Voting

In this method, each base classifier is trained on a random selection of a distinct subset of instances, labels, or features. The predictions of all base classifiers are then aggregated to generate the final prediction for label  $L_i$  by the Majority Vote. Given  $N$  base classifiers and a label  $L_i$ , the predicted value  $\tilde{y}_i$  for label  $L_i$  can be formulated by equation (1), where  $C_k(x_i)$  is the binary prediction of the  $k$ -th classifier for label  $L_i$ , and  $t$  is a decision threshold that creates a bipartition between relevant and irrelevant labels.

$$\tilde{y}_i = \begin{cases} 1, & \frac{1}{N} \sum_{k=1}^N C_k(x_i) \geq t \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

### B. Stacked Generalization

This strategy involves building a two-level stack of base classifiers: base-level and meta-level.  $N$  diverse classifiers are trained to predict probability distributions over all labels at the base level. Each classifier  $C_k$  generates a probability vector  $P_k$ . These probability distributions are then concatenated and used as input for a meta-classifier  $S(x_i)$ , which learns to combine the outputs of the base classifiers to generate the final multi-

label predictions for a new instance  $x_i$ , as defined in equation (2) :

$$\tilde{P}_{Li} = S \left( \bigcup_{k=1}^N P_{Lk} \right) \quad (2)$$

The meta-classifier  $S(x_i)$  produces a probability score for each label  $L_i$ , and a predefined threshold  $t$  is applied to determine the presence or absence of each label in the final prediction, as presented by equation (3).

$$\tilde{y}_i = \begin{cases} 1 & \text{if } \tilde{P}_{Li} \geq t \\ 0 & \text{Otherwise} \end{cases} \quad (3)$$

### C. Weighted Stacking

This approach extends traditional stacking by introducing label-specific weights  $w_{ik}$  that capture the relative importance of each label  $L_i$  in the classification process. The label weights are calculated using an *Average Performance Weighting* method, which evaluates each base classifier  $C_k$  on each label  $L_i$  using the *F1-Score* metric. These scores are normalized to ensure probabilistic coherence, as defined in equation (4).

$$W_{ik} = \frac{F1 - Score(C_k(L_i))}{\sum_{j=1}^N F1 - Score(C_j(L_i))} \quad (4)$$

The meta-classifier  $S(x_i)$  then computes the final probability  $\tilde{P}_{Li}$  for each label through weighted aggregation, as presented by equation (5), where  $P_{Lk}$  is the probability output assigned to label  $L_i$  by classifier  $C_k$ .

$$\tilde{P}_{Li} = S \left( \sum_{k=1}^N w_{ik} \times P_{Lk} \right) \quad (5)$$

For the final prediction, a fixed threshold  $t$  is applied such that label  $L_i$  is assigned to instance  $x_i$  if aggregated probability  $\tilde{P}_{Li}$  exceeds a threshold  $t$ . This selective thresholding ensures that only sufficiently confident predictions are retained.

## IV. EXPERIMENTAL DESIGN

This section presents the experimental multi-label datasets and the evaluation metrics used to assess the combined algorithms.

### A. Experimental Multi-Label Datasets

To evaluate our experimental study, we used five multi-label datasets from diverse domains (text, image, audio, video, and biology) listed in the Mulan repository: Medical [46], Flags [47], Birds [48], Mediamill [9], and Genbase [49]. Each dataset is described using basic meta-features. TABLE I provides an overview of the experimental datasets and their properties.

TABLE I. PROPERTIES OF MULTI-LABEL DATASETS USED IN OUR EXPERIMENTS.

Dataset	Domain	m	q	d	Card	avgIR	rDep
Madical	Text	978	45	1449	1.245	89.501	0.039
Flags	Image	194	7	19	3.392	2.255	0.381
Birds	Audio	645	19	260	1.014	5.407	0.123
Mediamill	Video	43910	101	120	4.376	256.405	0.342
Genbase	Biology	662	27	1186	1.252	37.315	0.157

### B. Multi-Label Evaluation Measures

In our experiments, performance was assessed using five evaluation metrics, including three example-based metrics (Hloss, Accuracy, and F1-score) defined by equations. (6) to (8), and two label-based metrics (Micro-F1, Macro-F1) defined by equations. (9) and (10), as summarized in TABLE II. For each metric, the test dataset comprises pairs  $(x_i, y_i)$ , where  $1 \leq i \leq n$ . Here,  $y_i \in \{0, 1\}$  denotes the ground-truth labels of the  $i$ -th test instance, while  $\tilde{y}_i = f(x_i)$  represents its predicted labels [24].

TABLE II. MULTI-LABEL EVALUATION METRICS USED IN OUR STUDY.

$$Hloss \downarrow = \frac{1}{N} \sum_{i=1}^N \frac{1}{Q} |h(x_i) \Delta Y_i| \quad (6)$$

$$Accuracy \uparrow = \frac{1}{N} \sum_{i=1}^N \frac{|h(x_i) \cap Y_i|}{|h(x_i) \cup Y_i|} \quad (7)$$

$$F1 - score \uparrow = \frac{1}{N} \sum_{i=1}^N \frac{2 \times |h(x_i) \cap Y_i|}{|h(x_i)| + |Y_i|} \quad (8)$$

$$Micro - F1 \uparrow = \frac{2 \times Micro_{precision} \times Micro_{recall}}{Micro_{precision} + Micro_{recall}} \quad (9)$$

$$Macro - F1 \uparrow = \frac{1}{M} \sum_{j=1}^M \frac{2 \times p_j \times r_j}{p_j + r_j} \quad (10)$$

## V. RESULTAS AND DISCUSION

This section presents and discusses the experimental results of the combined strategies. First, we analyze the performance of the EMLCs individually, including ECC, RAKEL, and RF-PCT, which were trained with their default parameter settings. Then, we evaluate their combined effectiveness using three different ensemble learning techniques: *Majority Voting*, *Stacking*, and *Weighted Stacking*, to assess the effectiveness of each combination strategy.

It is important to note that the performance evaluation of each method (individual or combined) must account for the complex characteristics of the benchmark datasets, such as *label dependencies*, *label imbalance*, and *high dimensionality*. This consideration is crucial, as each approach is designed to address specific challenges in MLC.

### A. Scenario 1: Performance Comparison of Individual Ensemble Methods

In this experiment, we evaluate ECC, RF-PCT, and RAKEL across five diverse datasets: Medical, Flag, Birds, Mediamill, and Genbase, using five metrics such as Hloss, Accuracy, F1-score, Micro-F1, and Macro-F1.

As shown in TABLE III, ECC achieved the best Hloss performance across all datasets, particularly for large-label datasets like Mediamill ( $q=101$ ). This superiority stems from ECC's ability to predict more labels correctly compared to the other methods. Additionally, RAKEL consistently outperformed RF-PCT in terms of Hloss for all datasets.

Regarding Accuracy metric, RAKEL outperformed the other methods on datasets with fewer labels (Flag, Birds, and Genbase). In contrast, ECC achieved higher accuracy on datasets with large-label datasets (Medical and Mediamill).

The F1-score metric is commonly used to evaluate the effectiveness of predicted label subsets for each instance, accounting for label dependencies. When label dependency is low—as observed in the Medical, Birds, and Genbase

datasets—ECC performs best, followed by RAKEL. This is because ECC effectively handles label dependencies by considering them sequentially, leading to accurate predictions in scenarios with weaker label interconnections. However, for the Mediamill and Flag datasets, which exhibits medium label dependency, RAKEL slightly outperforms ECC.

The F1-score (Macro and Micro) is considered more reliable than accuracy for evaluating model performance on imbalanced datasets. In terms of Micro-F1, ECC demonstrated strong performance on slightly imbalanced datasets, such as Birds (avgIR=5.407) and Flag (avgIR=2.255). Conversely, for Macro-F1, RAKEL outperformed other methods on highly imbalanced datasets like Mediamill (avgIR=256.405) and Medical (avgIR=89.501). This advantage stems from RAKEL's approach of dividing the output space into smaller label subsets for each base classifier, ensuring a more balanced label distribution.

Based on the above results, it can be concluded that no individual method consistently outperforms the others across all evaluation measures and datasets. Each method has its strengths and can be affected by the complex features of the tested dataset.

TABLE III. PERFORMANCE COMPARISON OF EMLCs FOR MLC ON VARIOUS DATASETS.

Dataset	EMLC Methods	Hloss	Accuracy	F1 score	Micro-F1	Macro-F1
Medical	ECC	0.010	0.765	0.799	0.815	0.363
	RF-PCT	0.012	0.753	0.781	0.786	0.330
	RAKEL	0.011	0.761	0.787	0.819	0.376
Flags	ECC	0.050	0.631	0.734	0.767	0.683
	RF-PCT	0.059	0.615	0.717	0.745	0.655
	RAKEL	0.054	0.633	0.742	0.761	0.684
Birds	ECC	0.041	0.114	0.178	0.443	0.242
	RF-PCT	0.046	0.148	0.123	0.416	0.210
	RAKEL	0.042	0.162	0.157	0.425	0.239
Mediamill	ECC	0.001	0.489	0.588	0.616	0.179
	RF-PCT	0.009	0.476	0.574	0.600	0.164
	RAKEL	0.006	0.486	0.598	0.618	0.233
Genbase	ECC	0.008	0.210	0.259	0.986	0.746
	RF-PCT	0.002	0.164	0.171	0.979	0.679
	RAKEL	0.009	0.234	0.225	0.979	0.747

The second phase of Scenario I evaluates ensemble method performance with respect to output space dimensionality ( $dim = m \times q \times d$ ). For this analysis, we organized the benchmark datasets in ascending order of dimensionality (TABLE IV) and conducted a comparative evaluation of all individual methods, measuring both training duration ( $T_{app}$ ) and inference time ( $T_{test}$ ).

According to obtained results in TABLE IV, it is evident that RF-PCT demonstrates superior performance across all benchmark datasets, particularly excelling in high-dimensional output spaces. This is due to its hierarchical structure, which reduces the output space by grouping labels and selecting attributes efficiently at each node of the tree, while taking into account interactions between labels.

### B. Scenario II: Comparative Analysis of Combined Approach Performance

The meta-model proposed in this scenario combines three ensemble methods: ECC, RAKEL, and RF-PCT, leveraging their complementary strengths to address key challenges in CML. RAKEL mitigates label imbalance by grouping rare

TABLE IV. TRAINING AND TEST TIME FOR INDIVIDUAL EMLCs FOR MLC ON VARIOUS DATASETS.

Dataset	TIMING	ECC	RF-PCT	RAKEL
Flag	T_app	5.18E-01	<b>3.92E-01</b>	5.54E-01
	T_test	2.14E-01	<b>1.31E-01</b>	1.62E-01
Birds	T_app	5.90E+00	<b>1.37E+00</b>	7.67E+00
	T_test	5.41E+00	<b>6.91E-01</b>	6.92E+00
Genbase	T_app	5.05E+00	<b>1.61E+00</b>	6.12E+00
	T_test	4.91E+00	<b>1.11E+00</b>	4.58E+00
Medical	T_app	3.45E+01	<b>2.34E+00</b>	3.32E+01
	T_test	3.34E+01	<b>2.91E+00</b>	3.20E+01
Mediamill	T_app	2.41E+04	<b>5.93E+02</b>	6.54E+03
	T_test	2.36E+04	<b>3.12E+02</b>	5.72E+03

labels with more frequent ones, ensuring better representation. ECC models label dependencies through sequential predictions, capturing complex interactions. While RF-PCT employs a hierarchical structure to organize labels into subsets, efficiently handling high-dimensionality while reducing model complexity.

These ensembles are combined using MV, ST, and WST strategies. As shown in TABLE V, stacking approaches exhibited strong robustness across most datasets, consistently achieved superior predictive accuracy. The enhanced performance of WST stems from its label-weighting mechanism, which prioritizes relevant labels during classification, enhancing decision-making precision. A key innovation of our approach involves employing RF-PCT as a meta-classifier in the stacking architecture. This hierarchical method effectively addresses the high-dimensionality challenge of the output space through its structured label organization. Furthermore, when the meta-classifier learns from the base models' predictions, it operates within an enriched information space that better captures complex label dependencies. The advantages generated by the hybrid sub-models contribute significantly to the overall system performance.

The obtained results validated two critical insights. First, the intrinsic characteristics of each tested dataset significantly influence the predictive performance of EMLC algorithms. This underscores the importance of considering these characteristics during method development and evaluation to ensure accurate and reliable outcomes. Second, integrating a label-weighting mechanism allows for assigning distinct priorities to labels during classification, thereby mitigating the influence of irrelevant ones. This refinement promotes more informed model decisions and enhances prediction accuracy. Additionally, the careful selection of heterogeneous and complementary base classifiers addresses the unique challenges of multi-label learning, as each method is specifically designed to solve particular problems within the application domain.

TABLE V. PERFORMANCE COMPARISON OF COMBINED APPROACHES.

Dataset	Ensemble Methods	Hloss	Accuracy	F1 Score	Micro F1↑	Macro F1↑
Medical	MV	0.011	0.705	0.733	0.658	0.631
	ST	0.004	0.781	0.649	0.827	0.742
	WST	0.003	0.789	0.664	0.841	0.771
Flag	MV	0.014	0.625	0.533	0.722	0.531
	ST	0.012	0.721	0.614	0.896	0.642
	WST	0.009	0.898	0.669	0.978	0.670
Birds	MV	0.025	0.466	0.661	0.402	0.212
	ST	0.018	0.575	0.691	0.461	0.393
	WST	0.012	0.618	0.721	0.647	0.639
Mediamill	MV	0.053	0.436	0.687	0.408	0.352
	ST	0.041	0.480	0.703	0.433	0.354
	WST	0.036	0.571	0.740	0.546	0.433
Genbase	MV	0.010	0.784	0.721	0.806	0.656
	ST	0.008	0.861	0.830	0.885	0.834
	WST	0.003	0.884	0.852	0.973	0.843

## VI. CONCLUSIONS

This study presents an in-depth evaluation of aggregation strategies used to build an ensemble methods for MLC, emphasizing their effectiveness across diverse benchmark datasets. The experimental results demonstrated that no single ensemble method consistently outperforms others across all evaluation metrics and datasets. Instead, each method exhibits strengths in specific scenarios depending on dataset characteristics such as *label dependencies*, *label imbalance*, and *high dimensionality*.

The results revealed that stacking-based approaches, particularly WST, provided the most robust performance across datasets. WST's advantage stems from its ability to dynamically assign importance to labels during classification, improving decision-making accuracy and mitigating the influence of irrelevant labels.

The findings demonstrate that optimal performance in MLC requires carefully tailored aggregation strategies that account for dataset-specific characteristics. The proposed hybrid approach, which strategically combines ECC, RAKEL, and RF-PCT through weighted stacking (WST), enhances classification robustness by leveraging their individual strengths, as each method addresses a specific challenge posed by the MLC field. Future work will focus on refining adaptive ensemble strategies that dynamically adjust to dataset complexities, further improving scalability and efficiency in large-scale multi-label learning application.

## REFERENCES

- [1] S. Guehria, H. Belleili, N. Azizi, et D. Zenakhra, « Boosting Multi-Label Classification Performance Through Meta-Model », *Int. J. Patt. Recogn. Artif. Intell.*, vol. 38, no 01, p. 2350033, janv. 2024, doi: 10.1142/S0218001423500337.
- [2] T. Joachims, « Text categorization with Support Vector Machines: Learning with many relevant features », in *Machine Learning: ECML-98*, Éd., in Lecture Notes in Computer Science, vol. 1398., Berlin, Heidelberg: Springer Berlin Heidelberg, 1998, p. 137-142. doi: 10.1007/BFb0026683.
- [3] B. Klimt et Y. Yang, « The Enron Corpus: A New Dataset for Email Classification Research », in *Machine Learning: ECML 2004*, vol. 3201, Éd., in Lecture Notes in Computer Science, vol. 3201., Berlin, 2004, p. 217-226. doi: 10.1007/978-3-540-30115-8\_22.
- [4] A. Clare et R. D. King, « Knowledge Discovery in Multi-label Phenotype Data », in *Principles of Data Mining and Knowledge Discovery*, vol. 2168, Éd., in Lecture Notes in Computer Science, vol. 2168., Berlin., 2001, p. 42-53. doi: 10.1007/3-540-44794-6\_4.
- [5] R. M. M. Vallim, « The Multi-label OCS with a Genetic Algorithm for Rule Discovery: Implementation and First Results ».
- [6] M. R. Boutell, J. Luo, X. Shen, et C. M. Brown, « Learning multi-label scene classification », *Pattern Recognition*, vol. 37, no 9, p. 1757-1771, sept. 2004, doi: 10.1016/j.patcog.2004.03.009.
- [7] S. Guehria, H. Belleili, N. Azizi, et S. B. Belhaouari, « "One vs All" Classifier Analysis for Multi-label Movie Genre Classification Using Document Embedding », in *Intelligent Systems Design and Applications*, vol. 1351, Éd., in Advances in Intelligent Systems and Computing, vol. 1351., Cham: Springer International Publishing, 2021, p. 478-487. doi: 10.1007/978-3-030-71187-0\_44.
- [8] W. Qu, Y. Zhang, J. Zhu, et Q. Qiu, « Mining Multi-label Concept-Drifting Data Streams Using Dynamic Classifier Ensemble », in *Advances in Machine Learning*, vol. 5828, Heidelberg, 2009, p. 308-321. doi: 10.1007/978-3-642-05224-8\_24.
- [9] C. G. M. Snoek, M. Worring, J. C. van Gemert, J.-M. Geusebroek, et A. W. M. Smeulders, « The challenge problem for automated detection of 101 semantic concepts in multimedia », in *Proceedings of the 14th annual ACM international conference on Multimedia - MULTIMEDIA '06*, Santa Barbara, CA, USA: ACM Press, 2006, p. 421. doi: 10.1145/1180639.1180727.
- [10] Hung-Yi Lo, Ju-Chiang Wang, Hsin-Min Wang, et Shou-De Lin, « Cost-Sensitive Multi-Label Learning for Audio Tag Annotation and Retrieval », *IEEE Trans. Multimedia*, vol. 13, no 3, p. 518-529, 2011, doi: 10.1109/TMM.2011.2129498.
- [11] I. Katakis, G. Tsoumakas, et I. Vlahavas, « Multilabel Text Classification for Automated Tag Suggestion », *Proceedings of ECML PKDD'08*, vol. vol 18, p. 75-83, 2008.
- [12] K. Ozonat et D. Young, Towards a universal marketplace over the web: statistical multi-label classification of service provider forms with simulated annealing. 2009, p. 1304. doi: 10.1145/1557019.1557158.
- [13] R. Rak, L. Kurgan, et M. Reformat, « A tree-projection-based algorithm for multi-label recurrent-item associative-classification rule generation », *Data & Knowledge Engineering*, vol. 64, no 1, p. 171-197, janv. 2008, doi: 10.1016/j.datak.2007.05.006.
- [14] B. Zhu et C. K. Poon, « Efficient Approximation Algorithms for Multi-label Map Labeling », in *Algorithms and Computation*, vol. 1741, in Lecture Notes in Computer Science, vol. 1741. Springer Berlin Heidelberg, 1999, p. 143-152. doi: 10.1007/3-540-46632-0\_15.
- [15] A. Rivoli, L. Parker, et A. de Carvalho, Food Truck Recommendation Using Multi-label Classification. 2017, p. 596. doi: 10.1007/978-3-319-65340-2\_48.
- [16] J. W. Kasubi et M. D. Huchaiah, « Human Activity Recognition for Multi-label Classification in Smart Homes Using Ensemble Methods », in *Artificial Intelligence and Sustainable Computing for Smart City*, Éd., in Communications in Computer and Information Science, vol. 1434., Cham: Springer International Publishing, 2021, p. 282-294. doi: 10.1007/978-3-030-82322-1\_21.
- [17] S. Guehria, H. Belleili, et N. Azizi, « A Survey on Ensemble Multi-label Classifiers », in *Proceedings of the 14th International Conference on Soft Computing and Pattern Recognition (SoCPaR 2022)*, vol. 648, A. Abraham, T. Hanne, N. Gandhi, P. Manghirmalani Mishra, A. Bajaj, et P. Siarry, Éd., in Lecture Notes in Networks and Systems, vol. 648., Cham: Springer Nature Switzerland, 2023, p. 100-109. doi: 10.1007/978-3-031-27524-1\_11.
- [18] G. Madjarov, D. Koccev, D. Gjorgjevikj, et S. Džeroski, « An extensive experimental comparison of methods for multi-label learning », *Pattern Recognition*, vol. 45, no 9, p. 3084-3104, sept. 2012, doi: 10.1016/j.patcog.2012.03.004.
- [19] X. Dong, Z. Yu, W. Cao, Y. Shi, et Q. Ma, « A survey on ensemble learning », *Front. Comput. Sci.*, vol. 14, no 2, p. 241-258, avr. 2020, doi: 10.1007/s11704-019-8208-z.
- [20] D. S. C. Nascimento, D. R. C. Bandeira, A. M. P. Canuto, et D. Araujo, « Investigating the Impact of Diversity in Ensembles of Multi-label Classifiers », in *2018 International Joint Conference on Neural Networks (IJCNN)*, Rio de Janeiro: IEEE, juill. 2018, p. 1-8. doi: 10.1109/IJCNN.2018.8489660.
- [21] A. Campagner, D. Ciucci, et F. Cabitza, « Aggregation models in ensemble learning: A large-scale comparison », *Information Fusion*, vol. 90, p. 241-252, févr. 2023, doi: 10.1016/j.inffus.2022.09.015.
- [22] V.-L. Nguyen, E. Hüllermeier, M. Rapp, E. L. Mencia, et J. Fürnkranz, « On Aggregation in Ensembles of Multilabel Classifiers », vol. 12323, 2020, p. 533-547. doi: 10.1007/978-3-030-61527-7\_35.
- [23] J. M. Moyano, E. L. Gibaja, K. J. Cios, et S. Ventura, « Combining multi-label classifiers based on projections of the output space using

- Evolutionary algorithms », *Knowledge-Based Systems*, vol. 196, p. 105770, mai 2020, doi: 10.1016/j.knsys.2020.105770.
- [24] S. Guehria, H. Belleili, et N. Azizi, « A Comparative Analysis of Ensemble Learning Methods for Multi-Label Classification on Bioinformatics », in 14th International Conference of Innovations in Bio-Inspired Computing and Applications, vol. 2, Cham: Springer International Publishing, 2023.
- [25] J. Read, B. Pfahringer, G. Holmes, et E. Frank, « Classifier Chains for Multi-label Classification », in *Machine Learning and Knowledge Discovery in Databases*, Éd., in *Lecture Notes in Computer Science*, vol. 5782, Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, p. 254-269. doi: 10.1007/978-3-642-04174-7\_17.
- [26] G. Tsoumakas et I. Vlahavas, « Random k-Labelsets: An Ensemble Method for Multilabel Classification », in *Machine Learning: ECML 2007*, Éd., in *Lecture Notes in Computer Science*, vol. 4701, Springer Berlin Heidelberg, 2007, p. 406-417. doi: 10.1007/978-3-540-74958-5\_38.
- [27] D. Kocev, C. Vens, J. Struyf, et S. Džeroski, « Ensembles of Multi-Objective Decision Trees », in *Machine Learning: ECML 2007*, Éd., in *Lecture Notes in Computer Science*, vol. 4701, Springer Berlin Heidelberg, 2007, p. 624-631. doi: 10.1007/978-3-540-74958-5\_61.
- [28] H. Blockeel, L. De Raedt, et J. Ramon, « Top-Down Induction of Clustering Trees », *Proc. 15th Intl. Conf. on Machine Learning*, déc. 2000.
- [29] J. Read, B. Pfahringer, et G. Holmes, « Multi-label Classification Using Ensembles of Pruned Sets », in 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy: IEEE, déc. 2008, p. 995-1000. doi: 10.1109/ICDM.2008.74.
- [30] G. Nasierding, A. Z. Kouzani, et G. Tsoumakas, « A Triple-Random Ensemble Classification Method for Mining Multi-label Data », in 2010 IEEE International Conference on Data Mining Workshops, Sydney, TBD, Australia: IEEE, déc. 2010, p. 49-56. doi: 10.1109/ICDMW.2010.139.
- [31] G. Madjarov, D. Gjorgjevikj, et S. Džeroski, « Dual Layer Voting Method for Efficient Multi-label Classification », in *Pattern Recognition and Image Analysis*, Éd., in *Lecture Notes in Computer Science*, vol. 6669, Springer Berlin Heidelberg, 2011, p. 232-239. doi: 10.1007/978-3-642-21257-4\_29.
- [32] Chen, W. Weng, S.-X. Wu, B.-H. Chen, Y.-L. Fan, et J.-H. Liu, « An efficient stacking model with label selection for multi-label classification », *Appl Intell*, vol. 51, no 1, p. 308-325, 2021, doi: 10.1007/s10489-020-01807-z.
- [33] W. Weng, C.-L. Chen, S.-X. Wu, Y.-W. Li, et J. Wen, « An Efficient Stacking Model of Multi-Label Classification Based on Pareto Optimum », *IEEE Access*, vol. 7, p. 127427-127437, 2019, doi: 10.1109/ACCESS.2019.2931451.
- [34] E. Loza Mencía et F. Janssen, « Learning rules for multi-label classification: a stacking and a separate-and-conquer approach », *Mach Learn*, vol. 105, no 1, p. 77-126, oct. 2016, doi: 10.1007/s10994-016-5552-1.
- [35] M. Kirchhof, L. Schmid, et C. Reining, « pRSL: Interpretable Multi-label Stacking by Learning Probabilistic Rules », 2021.
- [36] G. Tsoumakas, A. Dimou, E. Spyromitros, I. Kompatsiaris, et I. Vlahavas, « Correlation-Based Pruning of Stacked Binary Relevance Models for Multi-Label Learning », *Proceedings of the 1st International Workshop on Learning from Multi-Label Data*, p. 17, 2009.
- [37] H. Liu, Z. Wang, et Y. Sun, « Stacking model of multi-label classification based on pruning strategies », *Neural Comput & Applic*, vol. 32, no 22, p. 16763-16774, nov. 2020, doi: 10.1007/s00521-018-3888-0.
- [38] Y. Xia, K. Chen, et Y. Yang, « Multi-label classification with weighted classifier selection and stacked ensemble », *Information Sciences*, vol. 557, p. 421-442, mai 2021, doi: 10.1016/j.ins.2020.06.017.
- [39] S. Ghodrattnama et H. Abrishami Moghaddam, « Content-based image retrieval using feature weighting and C-means clustering in a multi-label classification framework », *Pattern Anal Applic*, vol. 24, no 1, p. 1-10, févr. 2021, doi: 10.1007/s10044-020-00887-4.
- [40] H. Wu, M. Han, Z. Chen, M. Li, et X. Zhang, « A Weighted Ensemble Classification Algorithm Based on Nearest Neighbors for Multi-Label Data Stream », *ACM Trans. Knowl. Discov. Data*, vol. 17, no 5, p. 72:1-72:21, févr. 2023, doi: 10.1145/3570960.
- [41] A. Büyükcakir, H. Bonab, et F. Can, « A Novel Online Stacked Ensemble for Multi-Label Stream Classification », in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*, Torino Italy: ACM, oct. 2018, p. 1063-1072. doi: 10.1145/3269206.3271774.
- [42] Y. Xia, K. Chen, et Y. Yang, « Multi-label classification with weighted classifier selection and stacked ensemble », *Information Sciences*, vol. 557, p. 421-442, mai 2021, doi: 10.1016/j.ins.2020.06.017.
- [43] N. Rastin, M. Taheri, et M. Z. Jahromi, « A stacking weighted k-Nearest neighbour with thresholding », *Information Sciences*, vol. 571, p. 605-622, sept. 2021, doi: 10.1016/j.ins.2021.05.030.
- [44] H. Wu, M. Han, Z. Chen, M. Li, et X. Zhang, « A Weighted Ensemble Classification Algorithm Based on Nearest Neighbors for Multi-label Data Stream », *ACM Trans. Knowl. Discov. Data*, p. 3570960, nov. 2022, doi: 10.1145/3570960.
- [45] M. A. Tahir, J. Kittler, et A. Bouridane, « Multilabel classification using heterogeneous ensemble of multi-label classifiers », *Pattern Recognition Letters*, vol. 33, no 5, p. 513-523, avr. 2012, doi: 10.1016/j.patrec.2011.10.019.
- [46] J. P. Pestian et al., « A shared task involving multi-label classification of clinical free text », in *Proceedings of the Workshop on BioNLP 2007 Biological, Translational, and Clinical Language Processing - BioNLP '07*, Prague, Czech Republic: Association for Computational Linguistics, 2007, p. 97. doi: 10.3115/1572392.1572411.
- [47] E. Gonçalves, A. Plastino, et A. Freitas, « A Genetic Algorithm for Optimizing the Label Ordering in Multi-label Classifier Chains. in 25th International Conference on Tools with Artificial Intelligence. 2013, p. 476. doi: 10.1109/ICTAI.2013.76.
- [48] F. Briggs et al., « The 9th annual MLSP competition: New methods for acoustic classification of multiple simultaneous bird species in a noisy environment », in 2013 IEEE International Workshop on Machine Learning for Signal Processing (MLSP), Southampton, United Kingdom: IEEE, sept. 2013, p. 1-8. doi: 10.1109/MLSP.2013.6661934.
- [49] S. Diplaris, G. Tsoumakas, P. A. Mitkas, et I. Vlahavas, « Protein Classification with Multiple Algorithms », in *Advances in Informatics*, Éd., in *Lecture Notes in Computer Science*, vol. 3746, Berlin, Heidelberg: Springer Berlin Heidelberg, 2005, p. 448-456. doi: 10.1007/11573036\_42.

# Adaptive Real-Time Scheduling Algorithms for Embedded Systems

1<sup>st</sup> Abdelmadjid Benmachiche 

Department of Computer Science

LIMA Laboratory

Chadli Bendjedid University

El-Tarf, PB 73, 36000, Algeria

benmachiche-abdelmadjid@univ-eltarf.dz


2<sup>nd</sup> Khadija Rais 

Informatics and Systems (LAMIS)

Echahid Cheikh Larbi Tebessi University Echahid Cheikh Larbi Tebessi University

Tebessa, 12002, Algeria

khadija.rais@univ-tebessa.dz

3<sup>rd</sup> Hamda Slimi 

Informatics and Systems (LAMIS)

Echahid Cheikh Larbi Tebessi University

Tebessa, 12002, Algeria

slimi.hamda@univ-tebessa.dz

**Abstract**—Embedded systems are becoming more in demand to work in dynamic and uncertain environments, and being confined to the strong requirements of real-time. Conventional static scheduling models usually cannot cope with runtime modification in workload, resource availability, or system updates. This brief survey covers the area of feedback-based control (e.g., Feedback Control Scheduling) and interdependence between tasks (e.g., Symbiotic Scheduling of Periodic Tasks) models. It also borders on predictive methods and power management, combining methods based on Dynamic Voltage and Frequency Scaling (DVFS). In this paper, key mechanisms are briefly summarized, influencing trade-offs relating to adaptivity/predictability, typical metrics of evaluation, and ongoing problems, especially in situations where safety is a critical factor, giving a succinct and easy-to-understand introduction to researchers and practitioners who have to cope with the changing environment of adaptive real-time systems.

**Index Terms**—Adaptive scheduling, real-time embedded systems, feedback control scheduling, symbiotic task scheduling, DVFS

## I. INTRODUCTION

The embedded systems that are used in modern times, from automated vehicles to smart medical equipment, are increasingly dynamic environments where the workloads, availability of resources, and external conditions vary dynamically. Conventional real-time scheduling algorithms, such as Rate-Monotonic (RM) and Earliest Deadline First (EDF), make an additional assumption that the set of tasks is known and that the execution times are fixed, where they cannot cope with uncertainty at runtime without making timing guarantees [6].

In autonomous systems, such as next-generation robots, adaptive path planning algorithms like hybrid PSO-APF approaches [2] demonstrate the need for real-time adaptation in dynamic environments with static and dynamic obstacles. Similarly, hybrid BFO/PSO algorithms for mobile robot navigation [17] showcase how bio-inspired optimization can enhance real-time decision-making in embedded systems.

To bridge this gap, adaptive real-time scheduling has gained significant traction, enabling systems to modify priorities, execution budgets, or resource allocations at runtime based on feedback, prediction, or environmental sensing. Among the promising directions are feedback control scheduling

(FCS) frameworks that use control-theoretic loops to maintain performance under load variations [16], and machine learning-enhanced predictive schedulers that leverage historical data to anticipate task behavior and proactively adjust schedules [8].

Recent research highlights this development, for instance Subramaniyan et al. proposed FC-GPU [22], a real-time embedded feedback control scheme that dynamically controls the allocation of GPU resources to soft deadline-based real-time edge AI applications, an essential feature of edge AI applications. In the meantime, Goksoy suggested a runtime monitoring system of ML-based schedulers that indicates a change of distribution during the execution of the task and allows a safe adjustment of safety-critical environments [7]. In another study [23], Zhao revealed a task-degradation-aware adaptive scheduler that is effective in managing the dynamically applications and maintaining the real-time restrictions.

This survey provides an introduction to adaptive real-time scheduling strategies of embedded systems, especially the feedback-based, predictive, and hybrid methods. We discuss how they can be integrated with system mechanisms such as DVFS, some of the most significant trade-offs (e.g., adaptivity vs. predictability), and the current gaps in verification and benchmarking.

## II. ADAPTIVE SCHEDULING APPROACHES

### A. Feedback-Based Scheduling

Adaptive scheduling reacts to real-time system performance metrics (e.g., deadline miss ratio, queue length, execution delays, or resource utilization) based on feedback to dynamically modify CPU/GPU allocation, task priorities, or execution budgets. Control-theoretic principles are used to determine the feedback-based reaction to the tactical choices made by the adaptive scheduling algorithm. In contrast to traditional methods of open-loop scheduling, which are based on offline worst-case execution time (WCET) measurements, feedback-based scheduling performs continual monitoring of system behavior and a corrective response in the form of control actions to ensure system stability, timing constraints, and quality of Service (QoS), and it does not require accurate a priori knowledge of workload.

One of the foundational structures in this area was presented by Lu et al. [16] with the concept of Feedback Control Scheduling (FCS), the application of classical control theory to real-time systems. Their work enabled the systematic modeling of computing systems as dynamical systems and the development of controllers to control ratios of CPU utilization and deadline misses. This paper showed that analytically tuned feedback controllers can offer high levels of transient and steady-state performance even in cases where task execution times change by large factors at runtime.

In another research paper by Subramaniyan and Wang [22], FC-GPU (Feedback Control GPU Scheduling) is introduced, the first feedback-based scheduling model specifically developed to fit GPUs in embedded and real-time systems. They use a multi-input, multi-output (MIMO) system to model the contention of resources in GPUs to extend feedback control scheduling to the heterogeneous architecture of a GPU. A MIMO controller is a dynamically adjusted task invocation rate controller, which adjusts the rate in response to measured response times. According to the experimental findings on NVIDIA RTX 3090 and AMD MI-100 GPUs, both show considerable enhancements in real-time and high-endurance performance under fluctuations in workloads during runtime.

Pan and Wei proposed a real-time workflow feedback scheduling system in cloud and distributed systems using deep reinforcement learning. They apply a regularized Deep Q-Network (R-DQN) that assigns workflow tasks to virtual machines depending on the condition of the system. This architecture proves to be more adaptive, scalable, and robust than conventional heuristic and fixed schedulers, which essentially serve as a smart feedback controller that learns the best scheduling policies online [18].

Scheduling of industrial processes has also been done successfully using feedback principles. He et al. [10] suggested a closed-loop gasoline blending scheduling scheme that unites real-time optimization and slack-based feedback. Their approach adds slack variables to the constraints of the processes and feeds actual deviations back into the scheduling model and re-optimizes production plans while accounting for actual deviations. Such a strategy enhances the similarity between theoretical scheduling and the real behavior of operations in a highly dynamic environment.

### *B. Symbiotic / Task-Coupling Scheduling*

Task-coupling or symbiotic scheduling is a higher-level approach to scheduling that takes advantage of inter-task dependencies and correlation/co-activation patterns in order to enhance system efficiency and predictability. This method, instead of handling tasks as individual entities, considers them cooperative or symbiotic. Cooperation and execution of tasks may allow for resolving resource contention while also improving cache and memory locality and end-to-end timing guarantees. The model works especially well in cyber-physical systems, multi-sensor hard workloads, and highly parallel architectures, in which the interactions among tasks significantly affect system-wide performance.

Posluns and Jeffrey [19], in their article about Symbiotic Task Scheduling and Data Prefetching, introduce a modern architectural view of the concept of symbiotic scheduling. They present the Task-Seeded Prefetcher (TSP) and Memory Response Task Scheduler (MRS), a joint hardware-software system, which allows task scheduling and memory subsystem behavior to cooperate. TSP acquires task-specific data access patterns and prefetches memory at the granularity of short-lived tasks, whereas MRS takes short-lived tasks as the source of prefetch status decisions. This symbiotic association achieves significant reductions in exposed DRAM latency and substantial performance gains in large-scale manycore processors.

Symbiotic scheduling was first investigated in the context of Simultaneous Multithreading (SMT) architectures in the form of the Symbiotic Job Scheduling (SOS) scheme by Snively and Tullsen [21]. SOS is a dynamic approach that finds workloads that run effectively as job co-scheduling combinations by sampling different combinations that may or may not work effectively. Through intelligent pairing of threads by using online sampling and a symbiosis-based scheduling phase, SOS proved to achieve better system throughput and lower response time by exploiting complementary usage of resources by threads.

Symbiotic concepts have been applied to real-time IoT workloads in distributed and edge-cloud environments. A semi-dynamic and real-time multiplexing algorithm of task scheduling in cloud-fog IoT systems was proposed by Abohamama et al. [1], where implicit task coupling is observed by optimization by means of permutation based on a modified genetic algorithm. The algorithm clusters and ranks tasks to maximize execution locality and resource fit between the fog and the cloud nodes, resulting in significant gains in makespan, latency, and failure rate for delay-sensitive IoT applications.

Recently, Kwon et al. [12] applied the concept of task-coupling to Industrial IoT (IIoT)-based flexible manufacturing systems that involve human-machine interaction. Their model considers operator-induced variation of tasks as explicit coupling events and pre-calculates executable joint resource strategies that combine CPU frequency scaling, memory allocation, and edge/cloud offloading choices. At runtime, the scheduler alternates between these coupled plans on the basis of perceived interactions, providing strict adherence to deadlines and optimizing energy use and system responsiveness.

### *C. Predictive / ML-Based Scheduling*

Predictive scheduling is a data-driven approach to scheduling that takes advantage of historical traces, machine telemetry, and real-time machine data to predict task performance, workload, and resource usage. It is in contrast to reactive scheduling mechanisms. Such methods are proactive by planning how systems will behave and can take measures that include dynamically scaling resources, reprioritizing tasks, or preemptive migration before situations arise in which a deadline is missed or performance is compromised. It is a powerful paradigm in very dynamic environments, such as cloud data centers,

edge computing platforms, the Internet of Things (IoT), and autonomous cyber-physical systems.

Pan and Wei [18] designed a representative cloud-based model and suggested a deep learning-based reinforcement learning scheduling system in real-time workflow applications. Their architecture uses a regularized Deep Q-Network (RDQN), which dynamically plans workflow tasks to virtual machine instances depending on the perceived system state. The approach performs well in optimizing workflows for workload uncertainty and increased resource usage in a heterogeneous workflow structure because it learns the best scheduling policies online, thus making it highly scalable.

In another study, Kesavan and his colleagues [11] proposed a model named Secure Edge-Enabled Multi-Task Scheduling (SEE-MTS) in edge- and IoE-based settings that combines reinforcement learning and security- and energy-conscious scheduling. Their system integrates edge computing and encrypted task management, dynamic key generation and verification mechanisms, and reinforcement learning to reduce task completion time and energy utilization. The framework is energy-efficient and offers strong security and assurance, which demonstrates the appropriateness of ML-based scheduling for safety-critical and resource-constrained distributed systems.

Conceptually and in terms of systems-level, Guo [9] explored the combination of neural networks and machine learning methods with real-time scheduling. This paper points out how neurodynamic systems and reinforcement learning can be used to solve constrained optimization problems subject to time constraints in real time, and the architectural and safety issues of determinism, hardware accelerators, and certification in safety-critical workloads. The research points out the significance of co-designing learning algorithms along with real-time system guarantees.

Related AI approaches in other domains include intrusion detection systems using deep learning [20] and voice recognition platforms using convolutional neural networks [3], which demonstrate the potential of ML techniques for real-time embedded applications. Furthermore, optimization techniques like bacterial foraging optimization (BFO) applied to Hidden Markov Models for speech recognition [4], [5] illustrate how adaptive algorithms can optimize system parameters in real-time applications.

Similarly, Gracias and Brooklyn [8] discussed AI and ML applications in predictive scheduling within real-time operating systems (RTOS). Their work uses supervised learning models to estimate the execution time of tasks, reinforcement learning to produce adaptive scheduling policies, and context-aware prioritization of tasks using neural networks. Their findings depict better latency, predictability, and throughput than conventional heuristic schedulers, particularly when dealing with extremely variable and unpredictable workloads.

#### D. Hybrid / DVFS-Integrated Scheduling

Hybrid scheduling approaches combine task management with Dynamic Voltage and Frequency Scaling (DVFS) for

the simultaneous optimization of power consumption, performance, and thermal limits in real-time embedded systems. These methods allow fine trade-offs between power consumption and deadline guarantees and jointly control the operating frequency and voltage of processing units, thus making them very suitable for battery-operated, resource-constrained, or thermally sensitive applications.

The researchers in [13] implemented the Flexible Invocation-Based Deep Reinforcement Learning (FiDRL) framework for DVFS scheduling in embedded systems. FiDRL enhances regular DRL methods by embedding the invocation timing of the agent within its action space, which allows the agent to self-invoke judiciously to minimize the overall energy of the system, accounting for the energy overhead of the DRL agent. FiDRL allows inter- and intra-task DVFS scheduling and also uses a hybrid on/off-chip algorithm to train and deploy on an embedded platform, achieving a 55.1% decrease in the energy overhead of agent invocation.

In another study [14], the authors proposed a real-time data-driven hybrid synchronization framework to address heterogeneous demand-capacity synchronization (HDCS) in flexible manufacturing systems. Their approach integrates hierarchical real-time data feedback with a ticket-enabled queuing mechanism (GiMS) to ensure seamless coordination between planning, scheduling, and execution. The framework combines global optimization models with local adaptive control mechanisms, enabling rapid reactions to disturbances while preserving overall system optimality.

For flexible and multiskilled manufacturing systems with unpredictable demand, Xinyi Li et al. [15] defined a real-time data-driven hybrid framework integrating planning, scheduling, and execution (PSE). A stratified real-time feedback data system with hybrid optimization is included in the framework to achieve efficiency and responsiveness of local and global performance simultaneously through the use of human and machine cooperation. Experimental results show cost-efficiency, timeliness, and good resource management, underscoring how hybrid scheduling with real-time data feedback optimizes energy, time, and operational performance in Industry 5.0.

### III. DISCUSSION

Adaptive real-time scheduling has evolved from a niche concept into a necessity for modern embedded systems operating in dynamic, uncertain, and resource-constrained environments. The surveyed approaches, feedback-based, symbiotic, predictive, and hybrid DVFS-integrated, each address different facets of runtime adaptability. Feedback control offers robustness through continuous system monitoring and corrective actions, making it suitable for environments with bounded but unpredictable perturbations. Symbiotic scheduling exploits structural relationships among tasks, improving predictability in cyber-physical and multi-sensor systems where tasks are inherently interdependent. Predictive and ML-based methods, particularly those leveraging deep reinforcement learning, promise high adaptability in complex, non-stationary workloads but often at the cost of determinism and explainability.

Meanwhile, hybrid DVFS-integrated strategies demonstrate that energy and timeliness can be co-optimized, especially in edge and battery-powered devices. However, all these paradigms share a common tension: the trade-off between adaptivity and predictability. While adaptivity enhances responsiveness to change, it inherently weakens the strong timing guarantees that traditional real-time systems rely on for safety certification. This tension becomes critical in domains like automotive or medical systems, where even minor violations of timing constraints can have severe consequences. Thus, the core challenge lies not in enabling adaptation, but in doing so safely and verifiably.

#### IV. RESEARCH GAPS

Despite significant progress, several critical gaps remain unresolved in the field of adaptive real-time scheduling. First, there is a notable lack of formal verification frameworks for adaptive schedulers. Most ML-based or feedback-driven approaches operate as black boxes, making it difficult to provide mathematical guarantees about schedulability under all possible runtime conditions. Second, current evaluation methodologies are fragmented and non-standardized; researchers use custom workloads, metrics, and platforms, which hinders fair comparison and reproducibility. Third, the overhead of adaptation mechanisms, whether computational, memory, or energy, is often underreported or assumed negligible, yet it can dominate system behavior in ultra-constrained embedded devices. Fourth, there is insufficient work on cross-layer co-design, where adaptation decisions jointly consider scheduling, memory management, thermal constraints, and communication (e.g., in networked embedded systems). Finally, most adaptive schedulers assume soft or firm real-time constraints; extending these techniques to hard real-time contexts, where deadlines must never be missed, remains largely unexplored, especially when adaptation itself introduces non-determinism.

#### V. FUTURE DIRECTIONS

Future research should focus on bridging the gap between intelligent adaptivity and rigorous real-time guarantees. One promising direction is the development of certifiable adaptive schedulers, where learning-based components are augmented with runtime monitors, fallback policies, or formal wrappers that enforce safety boundaries. Another avenue is the creation of open benchmarking suites for adaptive real-time systems, including standardized task sets, fault injection models, and metrics that capture both timing fidelity and adaptation efficiency. Additionally, integrating lightweight explainable AI (XAI) into predictive schedulers could improve trust and debuggability, enabling developers to understand why a scheduling decision was made and whether it aligns with system-level objectives. Furthermore, as embedded systems become increasingly heterogeneous (e.g., CPU-GPU-FPGA SoCs), future schedulers must support cross-architecture adaptation, dynamically partitioning workloads across processing elements while respecting end-to-end deadlines. Lastly, the emergence of human-in-the-loop embedded systems, such as

collaborative robots or assistive medical devices, demands schedulers that can adapt not only to computational load but also to human behavior, variability, and safety preferences, opening a new frontier in context-aware real-time adaptation.

#### VI. CONCLUSION

Adaptive real-time scheduling has become indispensable for modern embedded systems that must operate reliably in dynamic, uncertain, and resource-constrained environments. This survey has outlined four dominant paradigms: feedback-based control, which offers stability through runtime monitoring; symbiotic/task-coupling scheduling, which exploits inter-task relationships for improved predictability; predictive and machine learning-based methods, which enable proactive adaptation in complex workloads; and hybrid DVFS-integrated techniques, which jointly optimize energy efficiency and timeliness. While each approach brings unique strengths, they all grapple with the fundamental tension between adaptivity and predictability, especially critical in safety-critical domains where timing guarantees cannot be compromised.

Despite significant advances, key challenges remain, including the lack of standardized benchmarks, limited formal verification for learning-driven schedulers, and insufficient support for hard real-time guarantees under adaptation. Future work must bridge these gaps by developing certifiable, lightweight, and context-aware scheduling frameworks that combine the responsiveness of modern AI techniques with the rigor of classical real-time theory. As embedded systems grow increasingly intelligent and interconnected, adaptive scheduling will continue to evolve, not just as a performance enhancer, but as a foundational enabler of safe, efficient, and resilient real-time computing.

#### REFERENCES

- [1] Abdelaziz Said Abohamama, Amir El-Ghamry, and Eslam Hamouda. Real-time task scheduling algorithm for iot-based applications in the cloud-fog environment. *Journal of Network and Systems Management*, 30(4):54, 2022.
- [2] Abdelmadjid Benmachiche, Makhlof Dourdour, Moustafa Sadek Kahil, Mohamed Chahine Ghanem, and Mohamed Deriche. Adaptive hybrid pso-apf algorithm for advanced path planning in next-generation autonomous robots. *Sensors*, 25(18):5742, 2025.
- [3] Abdelmadjid Benmachiche, Bouzata Hadjar, Ines Boutabia, Ali Abdelatif Betouil, Majda Maatallah, and Amina Makhlof. Development of a biometric authentication platform using voice recognition. In *2022 4th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–7. IEEE, 2022.
- [4] Abdelmadjid Benmachiche and Amina Makhlof. Optimization of hidden markov model with gaussian mixture densities for arabic speech recognition. *WSEAS Transactions on Signal Processing*, 15:85–94, 2019.
- [5] Abdelmadjid Benmachiche, Amina Makhlof, and Tahar Bouhadada. Optimization learning of hidden markov model using the bacterial foraging optimization algorithm for speech recognition. *International Journal of Knowledge-Based and Intelligent Engineering Systems*, 24(3):171–181, 2020.
- [6] Arkajit Datta, Shamith D Rao, and CG Mohan. Adaptive real-time scheduler for embedded operating system. In *2nd Indian International Conference on Industrial Engineering and Operations Management*, 2022.
- [7] A Alper Goksoy, Alish Kanani, Satrajit Chatterjee, and Umit Ogras. Runtime monitoring of ml-based scheduling algorithms toward robust domain-specific socs. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 43(11):4202–4213, 2024.

- [8] Abram Gracias and Peter Brooklyn. Leveraging ai and ml for predictive scheduling in real-time operating systems. *Real-Time Systems*, 01 2025.
- [9] Zhishan Guo. When machine learning and neural networks marry real-time scheduling. *Real-Time Systems*, 61(2):320–325, 2025.
- [10] Renchu He, Xinyu Yan, Junjie Hua, Jiajiang Lin, and Liang Zhao. Closed-loop gasoline blending scheduling based on real-time optimized slack feedback. *Chemical Engineering Science*, 309:121426, 2025.
- [11] Thiruppathy Kesavan V, Venkatesan R, Wai Kit Wong, and Poh Kiat Ng. Reinforcement learning based secure edge enabled multi task scheduling model for internet of everything applications. *Scientific Reports*, 15(1):6254, 2025.
- [12] Gahyeon Kwon, Yeongeun Shim, Kyungwoon Cho, and Hyokyung Bahn. Real-time task scheduling and resource planning for iiot-based flexible manufacturing with human-machine interaction. *Mathematics*, 13(11):1842, 2025.
- [13] Jingjin Li, Weixiong Jiang, Yuting He, Qingyu Yang, Anqi Gao, Yajun Ha, Ender Özcan, Ruijin Bai, Tianxiang Cui, and Heng Yu. Fidlrl: Flexible invocation-based deep reinforcement learning for dvfs scheduling in embedded systems. *IEEE Transactions on Computers*, 2024.
- [14] Mingxing Li, Shiquan Ling, Ting Qu, Shan Lu, Ming Li, Daqiang Guo, Zhen He, and George Q Huang. Real-time data-driven hybrid synchronization for integrated planning, scheduling, and execution toward industry 5.0 human-centric manufacturing. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2025.
- [15] Xinyi Li, Ti Zhou, Haoyu Wang, and Man Lin. Energy-efficient computation with dvfs using deep reinforcement learning for multi-task systems in edge computing. *IEEE Transactions on Sustainable Computing*, 2025.
- [16] Chenyang Lu, John A Stankovic, Sang H Son, and Gang Tao. Feedback control real-time scheduling: Framework, modeling, and algorithms. *Real-Time Systems*, 23(1):85–126, 2002.
- [17] Amina Makhoul, Abdelmadjid Benmachiche, and Ines Boutabia. Enhanced autonomous mobile robot navigation using a hybrid bfo/pso algorithm for dynamic obstacle avoidance. *Informatica*, 48(17), 2024.
- [18] Jiahui Pan and Yi Wei. A deep reinforcement learning-based scheduling framework for real-time workflows in the cloud environment. *Expert Systems with Applications*, 255:124845, 2024.
- [19] Gilead Posluns and Mark C Jeffrey. Symbiotic task scheduling and data prefetching. In *Proceedings of the 58th IEEE/ACM International Symposium on Microarchitecture*, pages 140–155, 2025.
- [20] Brahim Khalil Sedraoui, Abdelmadjid Benmachiche, Amina Makhoul, and Chaouki Chemam. Intrusion detection with deep learning: A literature review. In *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–8. IEEE, 2024.
- [21] Allan Snaveley and Dean Tullsen. Symbiotic job scheduling for a simultaneous multithreading machine. *ACM SIGPLAN Notices*, 35:234–244, 09 2000.
- [22] Srinivasan Subramaniyan and Xiaorui Wang. Fc-gpu: Feedback control gpu scheduling for real-time embedded systems. *ACM Transactions on Embedded Computing Systems*, 24(5s):1–25, 2025.
- [23] Kaiyu Zhao, Jinming Wu, Yuan Zou, Xudong Zhang, and Tianyu Wang. Task-degradation aware adaptive dynamic scheduling for priority-based automotive cyber-physical systems. *IEEE Access*, 2024.

# Advancing a Sustainable and Adaptive Energy Future: Enabling Renewables Integration for Intelligent Power Management

1<sup>st</sup> Mohamed Salah Benkhalfallah

*Artificial Intelligence and Autonomous Things Laboratory*  
*Department of Mathematics and Computer Science*  
*University of Oum El Bouaghi*  
 Oum El Bouaghi, Algeria  
 mohamedsalah.benkhalfallah@univ-oeb.dz

2<sup>nd</sup> Sofia KOUAH

*Artificial Intelligence and Autonomous Things Laboratory*  
*Department of Mathematics and Computer Science*  
*University of Oum El Bouaghi*  
 Oum El Bouaghi, Algeria  
 sofia.kouah@univ-oeb.dz

**Abstract**—Energy is an essential requirement and a fundamental substrate of sustainable economic and societal development, underpinning industrial productivity, technological innovation, and human well-being. In response to the pressing global challenges of climate change, resource depletion, and energy insecurity, the transition from finite, carbon-intensive fossil fuels to renewable energy sources has become an imperative of the twenty-first century. This evolutionary shift is propelled by declining technology costs, international decarbonization commitments, and the urgent need to reduce greenhouse gas emissions. Renewable energy sources have emerged as indispensable components of the future energy landscape. This paper offers a comprehensive analysis of the pivotal role that renewable energy plays within intelligent energy management systems, emphasizing their capacity to optimize generation, distribution, and consumption through advanced digital technologies. Moreover, it investigates the transformative innovations and future trends that are accelerating the deployment of renewable energy, thereby fostering a sustainable, resilient, and inclusive energy ecosystem essential for preserving ecological integrity and ensuring intergenerational prosperity.

**Index Terms**—Renewable Energy, Intelligent Energy Management Systems, Sustainable Energy Systems, Energy Efficiency Optimization, Green Energy Integration, Artificial Intelligence, Internet of Things

## I. INTRODUCTION

Renewable energy sources have become a cornerstone in the paradigm shift toward intelligent energy management, offering an indispensable response to the pressing global challenges of climate change, energy insecurity, and unsustainable fossil fuel dependence. As the global energy sector undergoes a profound transformation, renewables, such as solar, wind, hydroelectric, and geothermal stand at the forefront of this transition, providing clean, inexhaustible, and environmentally benign alternatives that redefine conventional energy pathways [1], [2]. When synergistically integrated with intelligent energy management systems (IEMS), the latent potential

of these renewable sources is significantly amplified. Through the deployment of advanced monitoring, control, and optimization techniques, IEMS facilitate the efficient orchestration of energy generation, storage, distribution, and consumption, thereby enhancing grid reliability, system responsiveness, and environmental performance [3]. This research undertakes a comprehensive examination of the evolving role of renewable energy within the domain of intelligent energy management. It analyzes key technological enablers, system components, and advanced modeling approaches as discussed in the works of [4]–[6], and [7]. By situating this investigation within the broader context of digital transformation and energy decarbonization, the study aims to elucidate the transformative capacity of renewables in shaping a sustainable, technologically sophisticated, and resilient energy future. This paper is structured as follows: Section 2 provides the foundational background and contextual framework for the study. Section 3 offers a synthesis of the principal academic contributions within the domain. Section 4 delineates the essential components of intelligent renewable energy management systems, followed by an examination of the digital technologies that enable the seamless integration of renewable energy. Subsequent sections explore advanced storage solutions and articulate the multifaceted benefits associated with renewable energy adoption. The paper then addresses the key challenges impeding effective management, alongside an analysis of the policy, regulatory, and institutional frameworks that support renewable integration. Finally, the concluding section identifies and discusses emerging trends and future directions likely to influence the evolution of smart renewable energy systems.

## II. BACKGROUND

### A. Smart Energy Management

Smart energy management represents a transformative approach to optimizing energy systems through integrated control architectures, real-time monitoring networks, and data-driven resource allocation. Contemporary SEM frameworks rely on cyber-physical systems including IoT sensor networks, machine learning algorithms for predictive analytics, and distributed energy resource management platforms to achieve dynamic load balancing and demand optimization. These smart energy ecosystems deliver substantial efficiency gains while enabling seamless integration of variable renewable energy sources. Empirical studies show that implementing SEM can improve grid reliability indicators (SAIDI/SAIFI) by 30-40% while reducing energy waste through adaptive control strategies, positioning SEM as a pillar of the global energy transition [8].

### B. Sources of renewable energy

Renewable energy sources are a class of energy derived from natural renewal processes, whose depletion rates are minimal on an anthropogenic scale. These resources are distinguished by their intrinsic sustainability and significantly reduced environmental externalities compared to conventional fossil fuels. As illustrated in Figure 1, the portfolio of renewable energy sources includes [9]:

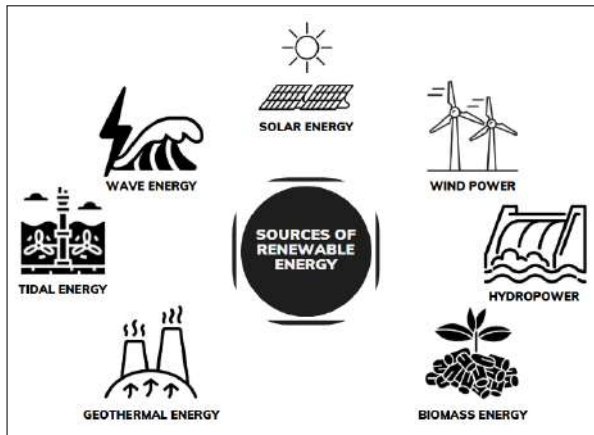


Fig. 1. Sources of renewable energy

- Solar energy: Photovoltaic (PV) systems convert incident solar radiation into electricity through the photovoltaic effect, while concentrating solar power plants use thermal energy to generate electricity through the Rankine cycle.
- Wind power: Horizontal and vertical axis wind turbines convert the kinetic energy of atmospheric

circulation into electrical energy through electromagnetic induction. Modern turbines achieve load factors over 50% in optimal locations.

- Hydropower: Potential energy from hydraulic load differences is converted into electricity by Francis, Kaplan, or Pelton turbines, while pumped-storage hydropower ensures essential grid flexibility.
- Biomass energy: Thermochemical and biochemical processes convert lignocellulosic biomass into syngas, biogas, or liquid biofuels, although sustainability demands rigorous management of raw materials.
- Geothermal energy: Deep-well systems exploit enthalpy gradients in hydrothermal reservoirs or enhanced geothermal systems, with binary-cycle power plants enabling efficient use at low temperatures.
- Tidal power: Axial-flow turbines and oscillating hydrofoils capture the kinetic energy of tidal currents, displaying predictable production patterns linked to lunar cycles.
- Wave energy: Point-absorbing buoys and oscillating water columns convert the swell motion of waves into electricity via hydraulic or direct-drive systems, although commercialization challenges remain.

The systemic integration of these renewable energy technologies is key to achieving net-zero emissions targets, with modeling suggesting they could supply 90% of the world's electricity by 2050 while reducing CO<sub>2</sub> emissions from the power sector by 70% compared with 2020 levels. Their modularity and distributed generation potential also enable resilient, decentralized energy architectures [10].

### C. Characteristics of renewable energies

Renewable energy sources have a set of characteristics that distinguish them from conventional fossil fuel systems and make them essential to the development of intelligent, sustainable energy infrastructures. These attributes not only support ecological and economic objectives but also enhance the adaptability of modern power systems [11], [12].

- Sustainability: Renewable energy is derived from natural sources such as solar radiation, wind, hydrological cycles, biomass, and geothermal energy. These resources are virtually inexhaustible on a human scale, enabling continuous energy production without degradation or depletion of the underlying resources.
- Environmental compatibility: Unlike fossil fuels, renewable energy technologies produce minimal greenhouse gas emissions and significantly lower levels of pollutants. Their deployment contributes to mitigating climate change, improving air and water

quality, and preserving biodiversity while reducing the ecological footprint of energy systems.

- **Resource diversity and geographic dispersion:** The heterogeneity of renewable resources enables geographically distributed production in a variety of environments. This diversity, ranging from solar and wind power to bioenergy and hydrothermal sources, facilitates the development of decentralized, resilient energy networks, less vulnerable to centralized outages and environmental disruptions.
- **Energy security and geopolitical stability:** By diversifying energy portfolios and reducing dependence on imported fossil fuels, renewable energies strengthen national energy autonomy. This contributes to greater resilience in the face of price volatility and geopolitical risks, reinforcing long-term energy security strategies.
- **Technological innovation and maturity:** Continuous advances in renewable energy technologies, supported by advances in materials science, control systems, AI optimization, and grid integration, have significantly improved their efficiency, reliability, and cost-effectiveness. These innovations are essential elements in the implementation of intelligent energy management systems and adaptive network architectures.
- **Cost-competitiveness:** The fall in the discounted cost of electricity for renewable energies, particularly solar photovoltaics, and onshore wind power, has made them economically competitive with, and in many regions cheaper than, traditional fossil-fuel generation. This economic viability is accelerating market adoption and stimulating investment in both developed and emerging economies.
- **Jobs and economic growth:** The renewable energy sector offers strong job creation opportunities, covering manufacturing, construction, operation, and maintenance. In addition, it supports local economies, fosters technological entrepreneurship, and stimulates innovation ecosystems aligned with green growth and just transition objectives.
- **Scalability and modularity:** Renewable energy systems can be deployed at a wide range of scales, from solar home panels and community microgrids to large-scale wind and solar farms. This modularity enables progressive development, tailored deployment strategies, and seamless integration into urban and rural contexts.
- **Energy access and rural development:** Off-grid and mini-grid renewable energy systems offer viable solutions for expanding access to electricity in remote and underserved areas. These systems foster rural development, improve quality of life, and support wider socio-economic transformation by providing

reliable, affordable, and clean energy services.

Collectively, these characteristics confirm the strategic importance of renewable energies in shaping a more sustainable, intelligent, and adaptive energy future. As technological convergence with digitization, AI, and IoT continues to evolve, the role of renewables will become increasingly essential in enabling smart energy management and meeting global decarbonization targets.

#### *D. Traditional vs. Smart Energy Infrastructure*

Traditional energy infrastructures are characterized by centralized production, unidirectional energy flows, and limited real-time monitoring, often resulting in inefficiencies, vulnerability to disruption, and lack of adaptability to demand fluctuations or decentralized energy integration. Conversely, smart energy infrastructures embody a paradigm shift towards decentralization, digitization, and intelligence. By harnessing advanced technologies such as IoT, AI, and two-way communication networks, smart systems enable real-time data acquisition, predictive analytics, dynamic load balancing, and the seamless integration of renewable energy sources. This transformation promotes greater grid resilience, operational efficiency, and sustainability, laying the foundations for a more adaptive, decentralized, and consumer-centric energy ecosystem in tune with the imperatives of the energy transition [13].

#### *E. Drivers of Renewable Integration*

The integration of renewable energies into modern electricity systems is being driven by a confluence of technological, environmental, economic, and political factors. Chief among these is the imperative to decarbonize the energy sector in the face of growing concerns about climate change, supported by international agreements. At the same time, rapid advances in digital technologies, such as smart grid architectures, AI-based forecasting, and energy storage systems, have greatly improved the feasibility and reliability of variable renewables. Economic factors, including the falling costs of solar, wind, and battery technologies, are further encouraging their large-scale deployment. In addition, evolving regulatory frameworks, green financing mechanisms and growing societal demand for clean, decentralized energy solutions are collectively catalyzing the systemic transition towards high-penetration integration of renewables into intelligent energy management systems [14].

### III. LITERATURE REVIEW

Numerous scholarly research has investigated the integration of diverse renewable energy sources within the domain of intelligent power management. The existing body of research has predominantly emphasized the global shift toward clean, sustainable, and environmentally responsible energy systems. This section presents

a comprehensive overview of the relevant literature, that has shaped current understanding and technological advancements in the integration of renewables for intelligent power management. The study [15] introduces a Railway Energy Management System (REMS) designed to optimize the operation of electric railway stations while reducing greenhouse gas emissions and operational costs. Utilizing mixed integer linear programming (MILP), the study integrates renewable energy resources (RERs), energy storage systems (ESSs), and regenerative braking energy (RBE) into a smart energy management framework. By considering the stochastic behavior of these energy sources and real-time data, the REMS significantly decreases daily operational costs, achieving cost reductions of up to 56.09% through efficient energy reuse and generation strategies. The findings underscore the effectiveness of combining smart grid architecture with renewable technologies in enhancing the sustainability of railway systems. The authors [8] provide a comprehensive review of smart energy management systems (SEMS) that integrate renewable energy sources, particularly focusing on solar power and the IoT. It evaluates various methodologies used in recent studies to optimize energy management, emphasizing the importance of efficient monitoring, data collection, and application management to meet rising energy demands sustainably. The authors analyze the roles of different stakeholders in these systems, present challenges and best practices, and highlight the critical need for innovative approaches to effective energy use. Furthermore, the paper identifies future research directions aimed at enhancing the performance and reliability of energy management systems in smart grid environments. Overall, the study seeks to contribute to the ongoing discourse on optimizing renewable energy utilization and improving system efficiencies in the context of smart technologies. The authors [16] presented a comprehensive study on an intelligent energy management system (EMS) solution for multiple renewable energy sources. The proposed system is designed to optimize the use of renewable energy sources by employing advanced technologies and flexible control mechanisms. The study highlighted the benefits of using an EMS, including significant energy savings and reduced carbon emissions. The technical solution is presented in detail, including the materials and methods considered, discussions regarding the proposed solution were presented, and some limitations of this system have also been identified. The study [17] delved into the integration of renewable energy sources into energy systems for smart cities, highlighting the importance of this strategy in achieving sustainable urban development. By employing an integrative literature methodology, the study collected and analyzed relevant publications. The analysis revealed the decisive roles of smart energy

systems in reducing CO<sub>2</sub> emissions, improving energy efficiency, and enhancing energy management. The study also explored the characteristics of integrated renewable energy systems based on solar, wind, geothermal, hydro, biomass, and waste sources, and identified existing problems and challenges for smart energy systems in the smart city. The study concludes that the deep and rapid penetration of renewable energy technologies can benefit modern society by creating a low-carbon economy and improving the quality of urban life. The paper [18], proposed a novel approach to efficiently manage energy from different sources and maintain a load-supply power balance in a renewable energy-based hybrid system. The proposed approach used a voting-based smart energy management system (VSEMS) that employs a rule-based energy management algorithm (EMA) to make decisions. The effectiveness of the proposed algorithm was verified through a case study analysis using a yearly usage profile, demonstrating its viability and effectiveness in energy management operations. It also emphasized the importance of reducing greenhouse gas emissions from conventional power plants and highlighted the role of renewable energy-based hybrid systems in achieving global emissions targets. The proposed approach increases customer participation in decision-making related to their energy supply and controls the intermittency of renewable energy sources efficiently. According to [19], an innovative IoT-based Intelligent Smart Energy Management System (ISEMS) is introduced, which uses advanced machine learning techniques to efficiently manage renewable energy sources without compromising user comfort. This system employs a user-configurable dynamic priority assignment feature and an accurate prediction model based on several machine-learning techniques. The proposed architecture was evaluated in a laboratory-level experimental set-up, which demonstrates an advanced SEMS system with an optimized load strategy and reliable communication. The system outperformed other prediction models due to its PSO-based SVM regression model, which shows significant improvement in results compared to state-of-the-art methods. The ISEMS system is also designed to handle energy demand in a smart grid environment with deep penetration of renewables, highlighting the importance of developing accurate renewable energy prediction models to manage demand-side appliances efficiently. The reviewed literature has played a pivotal role in advancing the field of intelligent energy management, particularly in the context of mitigating pressing climate challenges. These scholarly contributions have facilitated the efficient governance, conservation, and optimization of energy systems while enhancing accessibility and cost-effectiveness through the application of sophisticated AI methodologies, algorithms, and frameworks.

The studies encompass a wide array of energy-related domains, employing diverse technological approaches to address complex objectives. A comparative synthesis of the key strategies and methodologies adopted across these works is presented in Table I.

#### IV. COMPONENTS OF AN INTELLIGENT RENEWABLE ENERGY MANAGEMENT SYSTEM

An intelligent renewable energy management system is a sophisticated integration of hardware, software, and communication technologies designed to optimize the production, storage, distribution, and consumption of renewable energy on dynamic, decentralized power grids. While the architecture of these systems may vary according to scale, area of application, and regional requirements, several key components are essential to their successful operation [20]–[22].

- **Renewable energy infrastructure:** This fundamental layer includes technologies for the direct capture of renewable energy from natural sources. It includes photovoltaic panels, wind turbines, hydroelectric installations, biomass converters, and geothermal power stations, each contributing to diversified and decentralized energy production, adapted to the availability of local resources.
- **Sensor networks and data acquisition systems:** Smart systems rely heavily on a dense deployment of IoT sensors and smart meters to collect granular, real-time data on variables such as energy production, weather, temperature, and load demand. This data underpins the advanced analytics, system diagnostics, and adaptive control mechanisms essential to optimizing performance.
- **Energy storage systems:** To mitigate the intermittency of renewable energy sources, ESS components such as lithium-ion batteries, pumped storage systems, compressed air systems, and thermal storage units are used. These systems improve grid stability by providing load balancing, peak shaving, and time shifting of energy supply to match fluctuations in demand.
- **Energy management software platforms:** At the heart of IREMS functionality is intelligent software that orchestrates system operation using predictive algorithms, control strategies, and optimization models. These platforms support functionalities such as renewable energy production forecasting, demand response coordination, energy dispatch optimization, and outage detection, often relying on AI and machine learning for adaptive decision-making.
- **Communication and control infrastructure:** A robust, secure, and low-latency communication framework, including wireless sensor networks, fiber op-

tic links, standardized protocols, and edge computing nodes, facilitates seamless interaction between system components. This infrastructure guarantees synchronized operation, remote diagnostics, and real-time control of distributed energy resources.

- **Demand-side management modules:** Demand-side management strategies are integrated to influence consumer behavior and optimize load profiles through techniques such as dynamic pricing, automated demand response, and coordination of smart appliances. These measures reduce peak demand, improve energy efficiency, and support grid stability, particularly in contexts of high renewable energy penetration.
- **Grid integration and interconnection mechanisms:** For grid-connected deployments, the system must include bi-directional inverters, smart transformers, protective relays, and grid interface controllers to ensure safe and efficient interaction with the power grid. These components enable voltage regulation, frequency control, and fault management, contributing to a resilient and flexible grid architecture.
- **Monitoring and human-machine interfaces:** Advanced visualization tools, real-time dashboards, and mobile platforms provide operators and end-users with actionable information and control functionality. These interfaces support performance monitoring, fault diagnosis, system configuration, and user engagement, promoting transparency and user-centric energy governance.

Taken together, these components form the technological backbone of intelligent renewable energy management systems. Their synergetic integration enables decentralized coordination, improved energy reliability, and real-time optimization, essential pillars for achieving a sustainable, adaptive, and intelligent energy future.

#### V. DIGITAL TECHNOLOGIES ENABLING RENEWABLE ENERGY INTEGRATION

Digital technologies play a key role in the smooth integration of renewable energies into today's power systems, improving automation, coordination, and intelligence along the entire energy value chain. Innovations such as the IoT, AI, blockchain, and advanced data analytics facilitate real-time monitoring, predictive maintenance, decentralized energy control, and dynamic participation in the energy market. These technologies support the orchestration of decentralized energy resources (DER), optimize load forecasting and energy dispatch, and enable demand-side flexibility through smart meters and responsive appliances. In addition, digital twins and advanced computing enable grid operators to simulate system behavior and proactively respond to the fluctuations inherent in renewable energy production.

Collectively, these digital advances help foster an adaptive, resilient, and data-driven energy ecosystem capable of meeting the demands of a low-carbon future [23].

## VI. STORAGE SOLUTIONS FOR RENEWABLE ENERGY INTEGRATION

Storage solutions represent a foundational pillar for the optimal integration of intermittent renewable energy sources by addressing the temporal imbalances between energy production and consumption. Emerging technologies, such as lithium-ion batteries, green hydrogen storage, flywheel systems, and thermal energy storage, offer significant flexibility and grid stabilization capabilities, thereby enhancing the penetration of solar and wind power. However, their large-scale deployment remains contingent upon continued technological advancements, cost reductions, and the evolution of regulatory frameworks that incentivize investment. A critical assessment reveals that the effectiveness of these storage solutions varies based on geographical contexts and energy mix configurations, emphasizing the imperative for a systemic approach. This approach must holistically integrate storage technologies, demand-side management, and smart grid infrastructures to ensure a resilient, efficient, and sustainable energy transition [24], [25].

## VII. RENEWABLE ENERGY BENEFITS FOR INTELLIGENT POWER MANAGEMENT

The integration of renewable energies into smart energy management systems offers several strategic, environmental, and economic benefits that underpin the transformation to sustainable, smart electricity systems. These benefits go beyond decarbonization and encompass system efficiency, resilience, and long-term viability [26], [27].

- **Environmental sustainability:** Renewable energy sources such as solar, wind, hydro, and geothermal emit negligible greenhouse gases during operation, offering a low-carbon alternative to conventional fossil fuel systems. When integrated into SEMS, these clean sources enable emissions to be monitored in real-time and distributed in an optimized way, helping to mitigate climate change, improve air quality, and meet global sustainable development targets.
- **Improved energy efficiency:** Smart systems exploit digital technologies, including real-time data analysis, predictive algorithms, and AI-driven control mechanisms, to optimize energy flows between generation, storage, and consumption. This synergy reduces transmission losses, maximizes resource utilization, and promotes adaptive energy planning,

significantly improving the overall efficiency of renewable electricity networks.

- **Cost optimization and economic viability:** The rapid decline in the discounted cost of renewable energy technologies, particularly solar photovoltaic and wind power, has made them increasingly competitive and often cheaper than fossil fuel alternatives. Combined with intelligent energy management techniques such as demand side management, peak load shaving, and storage integration, SEMS offers significant operating cost savings while minimizing dependence on volatile fuel markets.
- **Energy autonomy and security:** The localized nature of renewable energy resources enhances energy sovereignty by reducing dependence on imported fuels and improving resilience to geopolitical and market disruptions. Intelligent management platforms support this autonomy by enabling dynamic load control, resource forecasting, and self-healing capabilities in decentralized energy environments.
- **Grid flexibility and resilience:** SEMS integrating renewable energies improves grid adaptability by enabling distributed generation, modular system design, and real-time response to fluctuations in supply and demand. Combined with storage systems and advanced control infrastructure, these capabilities enhance the grid's ability to withstand disruptions, manage intermittency, and recover from disruptions, whether due to natural disasters, cyber threats, or equipment failure.
- **Socio-economic development and job creation:** The expansion of the renewable energy sector is boosting employment in a variety of areas, including technology manufacturing, infrastructure deployment, maintenance, and digital services. This growth not only supports economic development but is also part of the wider goals of a just energy transition by promoting green entrepreneurship and inclusive participation in energy markets.

As the global energy landscape evolves towards decarbonization and decentralization, the convergence of renewables and smart management systems will form the cornerstone of future energy architectures. Their integration is essential to create reliable, adaptive, and sustainable energy ecosystems capable of meeting current and future demands [26].

## VIII. RENEWABLE ENERGY MANAGEMENT CHALLENGES

Integrating and managing renewable energies within modern energy systems presents a complex set of technical, infrastructural, economic, and socio-political challenges that need to be systematically addressed to facilitate a smart, sustainable energy transition. One

of the main problems lies in the intermittency and variability of renewable energy sources such as solar and wind power, whose production is intrinsically dependent on meteorological and diurnal conditions. This temporal inconsistency complicates the real-time balancing of supply and demand, requiring the deployment of advanced forecasting algorithms, flexible generation assets, and dynamic scheduling protocols [28]. Grid integration poses another major challenge, as renewable energy generation is predominantly decentralized and geographically dispersed. Existing grid infrastructures, originally designed for centralized fossil-fuel generation, require major upgrades, including the implementation of smart grid technologies, two-way communication systems, and flexible transmission frameworks, to guarantee grid stability and facilitate smooth penetration of renewable energies. Added to this is limited grid capacity, particularly in regions where infrastructure development lags behind renewable energy deployment. Large-scale transmission and distribution network upgrades are not only capital-intensive but also constrained by regulatory, environmental, and land-use obstacles [29]. Energy storage technologies are essential for mitigating intermittency and improving grid reliability, but they remain economically and technically limited. Although advances in lithium-ion batteries, flow batteries, and other storage modes are promising, their large-scale, cost-effective deployment remains a major hurdle. The increasing digitization of renewable energy systems also introduces new cybersecurity vulnerabilities, as interconnected assets and control platforms become potential targets for cyberattacks, requiring robust cybersecurity strategies and resilient system architectures [30]. In addition, political and regulatory uncertainties often hamper the scalability of renewable energy projects. Inconsistent incentives, bureaucratic permitting procedures, and fragmented energy governance structures can delay project implementation and discourage private investment. At the same time, financial and economic constraints, including high initial investment costs and the perceived risks associated with emerging technologies, underline the need for innovative financing mechanisms, risk mitigation strategies, and market-driven frameworks to ensure long-term economic viability [29]. Developing human capital is another crucial issue, as the renewable energy sector requires a skilled workforce with interdisciplinary expertise in engineering, data science, and systems management. Investment in technical education and vocational training is essential to sustain the sector's growth and innovation. Finally, public acceptance plays a crucial role in the deployment of renewable energies. Societal concerns about land use, aesthetic impacts, noise pollution, and ecological disruption need to be addressed through transparent stakeholder engagement, en-

vironmental assessments, and inclusive decision-making processes [28]. Despite these multifaceted challenges, rapid technological progress, favorable market trends, and global climate imperatives make renewable energies a mainstay of future energy systems. Overcoming these obstacles through coordinated policies, innovation, and infrastructure development will be key to achieving a smart, resilient, and low-carbon energy future [30].

#### IX. POLICY AND REGULATORY FRAMEWORKS FOR RENEWABLE ENERGY ADOPTION

Policy and regulatory frameworks play a decisive role in the acceleration and deployment of renewable energies, shaping the incentives, mandates, and market mechanisms essential to their effective integration into modern energy systems. These frameworks encompass a wide range of instruments, such as feed-in tariffs, renewable portfolio standards, tax subsidies, carbon pricing, and technical standards, designed to lower economic and institutional barriers while stimulating technological innovation and investment. To be effective, policies must be coherently formulated to harmonize climate mitigation objectives, energy security imperatives, and socio-economic priorities, while remaining adaptable to local conditions and global dynamics, including the commitments set out in the Paris Agreement. A comprehensive analysis of these mechanisms reveals their differentiated impact on the pace of energy transition, distributive equity, and systemic resilience, underlining the crucial need for multi-level, participatory, and adaptive governance capable of supporting a just and sustainable energy future [31], [32].

#### X. CONCLUSION AND FUTURE PROSPECTS

Embracing a sustainable energy future through the integration of renewable energy sources and intelligent energy management systems represents a pivotal trajectory toward the realization of a cleaner, more efficient, and resilient global energy infrastructure. This study has underscored the fundamental role of renewables in smart energy management, emphasizing how their increasing penetration presents transformative opportunities for optimizing energy generation, distribution, and consumption. Harnessing renewable energy not only mitigates climate change by significantly reducing greenhouse gas emissions and environmental pollutants but also fosters long-term environmental stewardship. Intelligent management systems, driven by sophisticated technologies and autonomous decision-making algorithms, enhance energy efficiency, facilitate supply-demand equilibrium, and bolster grid flexibility and stability [33]. Moreover, the adoption of renewable energy in conjunction with smart energy management delivers economic benefits such as cost reduction and employment generation,

while empowering individuals, communities, and nations to become active agents in the energy transition [34]. By collectively advancing toward a sustainable future, we can unlock the full potential of renewable resources, ensuring universal access to clean, reliable, and affordable energy while safeguarding ecological integrity for future generations [35]. Looking ahead, the integration of renewable energy within smart energy management is poised for considerable evolution. The continued expansion and growing competitiveness of technologies such as solar photovoltaics and wind turbines are expected to substantially increase their share in the global energy portfolio. Breakthroughs in energy storage, including next-generation batteries and green hydrogen systems, will address intermittency challenges and facilitate deeper integration of variable renewables. Concurrently, the application of AI and advanced data analytics will enhance predictive accuracy, operational optimization, and strategic energy planning. The proliferation of IoT devices and enhanced connectivity will enable real-time monitoring and adaptive control of energy flows, while decentralized energy architectures and peer-to-peer trading platforms powered by blockchain technologies will decentralize energy governance and empower prosumers. Furthermore, innovations such as grid-interactive buildings, resilient microgrids, and the electrification of transport will reinforce renewable integration and grid robustness. Collectively, these emerging trends underscore the transformative potential of renewables in shaping an intelligent, decentralized, and sustainable energy paradigm [36].

# REFERENCES

- [1] B. Hemavathi, G. Vidya, A. KS, *et al.*, "Machine learning in the era of smart automation for renewable energy materials," *e-Prime-Advances in Electrical Engineering, Electronics and Energy*, vol. 7, p. 100458, 2024.
- [2] X. Yuan, C.-W. Su, M. Umar, X. Shao, and O.-R. LobonȚ, "The race to zero emissions: Can renewable energy be the path to carbon neutrality?," *Journal of Environmental Management*, vol. 308, p. 114648, 2022.
- [3] S. A. Qadir, H. Al-Motairi, F. Tahir, and L. Al-Fagih, "Incentives and strategies for financing the renewable energy transition: A review," *Energy Reports*, vol. 7, pp. 3590–3606, 2021.
- [4] M. R. AT, B. Balaji, S. A. P. RR, R. C. Naidu, R. Kumar, P. Ramachandran, S. Rajkumar, V. N. Kumar, G. Aggarwal, and A. M. Siddiqui, "Intelligent energy management across smart grids deploying 6g iot, ai, and blockchain in sustainable smart cities," *IoT*, vol. 5, no. 3, pp. 560–591, 2024.
- [5] M. M. Khayyat and B. Sami, "Energy community management based on artificial intelligence for the implementation of renewable energy systems in smart homes," *Electronics*, vol. 13, no. 2, p. 380, 2024.
- [6] Z. Yang, "Renewable energy management in smart grid with cloud security analysis using multi agent machine learning model," *Computers and Electrical Engineering*, vol. 116, p. 109177, 2024.
- [7] M. Algafr, A. Alghazi, Y. Almoghathawi, H. Saleh, and K. Al-Shareef, "Smart city charging station allocation for electric vehicles using analytic hierarchy process and multiobjective goal programming," *Applied Energy*, vol. 372, p. 123775, 2024.
- [8] C. K. Rao, S. K. Sahoo, and F. F. Yanine, "A literature review on an iot-based intelligent smart energy management systems for pv power generation," *Hybrid advances*, vol. 5, p. 100136, 2024.
- [9] D. P. Kothari, R. Ranjan, and K. Singal, "Renewable energy sources and emerging technologies," 2021.
- [10] S. Bouckaert, A. F. Pales, C. McGlade, U. Remme, B. Wanner, L. Varro, D. D'Ambrosio, and T. Spencer, "Net zero by 2050: A roadmap for the global energy sector," 2021.
- [11] S. Afrane, J. D. Ampah, and E. M. Aboagye, "Investigating evolutionary trends and characteristics of renewable energy research in africa: a bibliometric analysis from 1999 to 2021," *Environmental Science and Pollution Research*, vol. 29, no. 39, pp. 59328–59362, 2022.
- [12] J. Xu, T. Lv, X. Hou, X. Deng, N. Li, and F. Liu, "Spatiotemporal characteristics and influencing factors of renewable energy production in china: A spatial econometric analysis," *Energy Economics*, vol. 116, p. 106399, 2022.
- [13] M. W. Khan, S. Saad, S. Ammad, K. Rasheed, and Q. Jamal, "Smart infrastructure and ai," in *AI in Material Science*, pp. 193–215, CRC Press, 2024.
- [14] V. Anbumozhi and K. S. Bhupendra, *Cross-border integration of renewable energy systems: experiences, impacts, and drivers*. Taylor & Francis, 2024.
- [15] M. Davoodi, H. Jafari Kaleybar, M. Brenna, and D. Zaninelli, "Sustainable electric railway system integrated with distributed energy resources: Optimal operation and smart energy management system," *International Transactions on Electrical Energy Systems*, vol. 2025, no. 1, p. 6626245, 2025.
- [16] N. C. Gaitan, I. Ungurean, G. Corotinschi, and C. Roman, "An intelligent energy management system solution for multiple renewable energy sources," *Sustainability*, vol. 15, no. 3, p. 2531, 2023.
- [17] A. T. Hoang, X. P. Nguyen, *et al.*, "Integrating renewable sources into energy system for smart city as a sagacious strategy towards clean and sustainable process," *Journal of cleaner production*, vol. 305, p. 127161, 2021.
- [18] S. Bhattacharjee and C. Nandi, "Design of a voting based smart energy management system of the renewable energy based hybrid energy system for a small community," *Energy*, vol. 214, p. 118977, 2021.
- [19] P. Pawar, M. TarunKumar, *et al.*, "An iot based intelligent smart energy management system with accurate forecasting and load strategy for renewable generation," *Measurement*, vol. 152, p. 107187, 2020.
- [20] F. Villano, G. M. Mauro, and A. Pedace, "A review on machine/deep learning techniques applied to building energy simulation, optimization and management," *Thermo*, vol. 4, no. 1, pp. 100–139, 2024.
- [21] K. Parvin, M. Hannan, L. H. Mun, M. H. Lipu, M. G. Abdolrasol, P. J. Ker, K. M. Muttaqi, and Z. Dong, "The future energy internet for utility energy service and demand-side management in smart grid: Current practices, challenges and future directions," *Sustainable Energy Technologies and Assessments*, vol. 53, p. 102648, 2022.
- [22] D. Mariano-Hernández, L. Hernández-Callejo, A. Zorita-Lamadrid, O. Duque-Pérez, and F. S. García, "A review of strategies for building energy management system: Model predictive control, demand side management, optimization, and fault detect & diagnosis," *Journal of Building Engineering*, vol. 33, p. 101692, 2021.
- [23] B. Lin and C. Huang, "Promoting variable renewable energy integration: the moderating effect of digitalization," *Applied Energy*, vol. 337, p. 120891, 2023.
- [24] M. M. Rana, M. Uddin, M. R. Sarkar, S. T. Meraj, G. Shafiullah, S. Mueen, M. A. Islam, and T. Jamal, "Applications of energy storage systems in power grids with and without renewable energy integration—a comprehensive review," *Journal of energy storage*, vol. 68, p. 107811, 2023.
- [25] K. M. Tan, T. S. Babu, V. K. Ramachandaramurthy, P. Kasiathan, S. G. Solanki, and S. K. Raveendran, "Empowering smart grid: A comprehensive review of energy storage technology and

- application with renewable energy integration,” *Journal of Energy Storage*, vol. 39, p. 102591, 2021.
- [26] Z. Liu, Y. Sun, C. Xing, J. Liu, Y. He, Y. Zhou, and G. Zhang, “Artificial intelligence powered large-scale renewable integrations in multi-energy systems for carbon neutrality transition: Challenges and future perspectives,” *Energy and AI*, vol. 10, p. 100195, 2022.
  - [27] N. Mostafa, H. S. M. Ramadan, and O. Elfarouk, “Renewable energy management in smart grids by using big data analytics and machine learning,” *Machine Learning with Applications*, vol. 9, p. 100363, 2022.
  - [28] M. S. Benkhalfallah, S. Kouah, and M. Ammi, “Smart energy management systems,” in *Novel & Intelligent Digital Systems Conferences*, pp. 1–8, Springer, 2023.
  - [29] M. Tvaronavičienė, “Towards renewable energy: opportunities and challenges,” *Energies*, vol. 16, no. 5, p. 2269, 2023.
  - [30] S. T. Meraj, S. S. Yu, M. S. Rahman, K. Hasan, M. H. Lipu, and H. Trinh, “Energy management schemes, challenges and impacts of emerging inverter technology for renewable energy integration towards grid decarbonization,” *Journal of Cleaner Production*, vol. 405, p. 137002, 2023.
  - [31] A. Kylili, Q. Thabit, A. Nassour, and P. A. Fokaides, “Adoption of a holistic framework for innovative sustainable renewable energy development: A case study,” *Energy sources, Part A: Recovery, utilization, and environmental effects*, vol. 47, no. 1, pp. 6157–6177, 2025.
  - [32] C. E. Hoicka, J. Lowitzsch, M. C. Brisbois, A. Kumar, and L. R. Camargo, “Implementing a just renewable energy transition: Policy advice for transposing the new european rules for renewable energy communities,” *Energy Policy*, vol. 156, p. 112435, 2021.
  - [33] M. S. Benkhalfallah, S. Kouah, and F. Benkhalfallah, “Enhancing advanced time-series forecasting of electric energy consumption based on rnn augmented with lstm techniques,” in *Artificial Intelligence and Its Practical Applications in the Digital Economy* (Y. Mohamed Elhadj, M. Farouk Nanne, A. Koubaa, F. Meziane, and M. Deriche, eds.), Springer Cham, 2024.
  - [34] M. Saleem, “Possibility of utilizing agriculture biomass as a renewable and sustainable future energy source,” *Heliyon*, vol. 8, no. 2, 2022.
  - [35] L. Li, J. Lin, N. Wu, S. Xie, C. Meng, Y. Zheng, X. Wang, and Y. Zhao, “Review and outlook on the international renewable energy development,” *Energy and Built Environment*, vol. 3, no. 2, pp. 139–157, 2022.
  - [36] A. Azarpour, O. Mohammadzadeh, N. Rezaei, and S. Zendeheboudi, “Current status and future prospects of renewable and sustainable energy in north america: Progress and challenges,” *Energy Conversion and Management*, vol. 269, p. 115945, 2022.

TABLE I  
COMPARATIVE ANALYSIS OF SOME SCIENTIFIC STUDIES

Study	Strategy applied	Renewable energy sources							Objectives
		Solar Energy	Wind Energy	Hydropower	Biomass Energy	Geothermal Energy	Tidal Energy	Wave Energy	
[15]	Application of MILP strategy to optimize the energy management of electric railway stations.	✓	✓						Optimization of energy management, cost reduction, integration of renewable energy, real-time data utilization, and surplus energy management.
[8]	Provide an interpretive analysis of current methodologies, highlighting gaps and suggesting directions for future research.	✓	✓	✓					Evaluate energy management strategies, analyze IoT integration, identify best practices, highlight future directions, and foster knowledge development.
[16]	Design, develop, and test an intelligent system that integrates multiple sources of renewable energy.	✓	✓	✓					Achieve maximum efficiency in the energy management system.
[17]	Integrate renewable resources into the smart city energy system.	✓	✓	✓	✓	✓		✓	Reducing CO <sub>2</sub> emissions, improving energy efficiency, and enhancing its management. Achieve more sustainable, smarter, and cleaner cities in the future.
[18]	Propose the design of a VSEMS.	✓	✓		✓				Efficiently manage energy from different sources. Maintain a balance between energy supply and demand.
[19]	Propose an ISEMS architecture for demand-side energy management considering a renewable source.	✓							Enhance real-time energy systems management. Manage demand-side devices efficiently.

# Attention U-NET for Medical Image Segmentation

Meriem Chibani

Artificial Intelligence and Autonomous Things Laboratory

Department of Mathematics and computer Science

University of Oum El Bouaghi, Oum El Bouaghi, Algeria

Email: meriem.chibani@univ-ueb.dz

**Abstract** - Medical image segmentation is crucial for accurate disease diagnosis and analysis. Deep learning has shown great promise in this field by eliminating the need for manual feature engineering and delivering strong results in tasks such as disease classification and image segmentation. However, two key challenges continue to restrict performance in medical imaging. First, the region of interest typically occupies only a small portion of the image, with much of the remaining area being irrelevant. Second, target structures can appear at different scales across the input data. These factors can mislead traditional classification and segmentation models, reducing their effectiveness. To overcome these challenges, attention mechanisms have been introduced to help models focus on the most relevant parts of the image while ignoring uninformative regions. In this study, we utilize the Attention U-Net architecture, which integrates Attention Gates (AGs) to automatically concentrate on target structures that may vary in shape and size. These AGs eliminate the need for separate tissue or organ localization components, often required in cascaded convolutional neural networks (CNNs), by enabling the model to suppress irrelevant features and enhance critical ones relevant to the task. Attention Gates can be seamlessly integrated into existing CNN architectures like U-Net with minimal computational overhead, significantly boosting both prediction accuracy and model sensitivity. We validated the effectiveness of the Attention U-Net through experiments on a skin cancer dataset. Evaluation metrics, including accuracy and loss, demonstrated strong model performance, confirming its potential and applicability in the medical imaging field.

**Index Terms** - Deep learning, medical image, Attention U-net, semantic segmentation.

## I. INTRODUCTION

Semantic segmentation is one of the high-level activities, that leads to full scene interpretation. The growing number of applications that rely on inferring information from images emphasizes the significance of scene understanding as a fundamental computer vision problem. Self-driving cars, virtual reality, and human-computer interaction are a few of these applications. Many semantic segmentation problems are being tracked using deep architectures, most frequently convolutional neural nets, which should outperform alternative methods by a significant margin in terms of accuracy and efficiency due to the current surge in popularity of deep learning [1].

Attention networks were initially introduced in machine translation [10] to help models focus on relevant parts of the input when generating each word in the output. They were then modified for image-related tasks like image captioning [3], in which the model learns to focus on particular areas of

an image while producing words that correlate to those areas. For image analysis, several kinds of soft attention mechanisms have been created. In order to capture non-linear dependencies, squeeze-based attention [4], [5] compresses information in both spatial and channel dimensions. The resulting summary is then used as a weighting tensor. Contrarily, correlation-based attention [6] looks for connections between different parts, either across channels or spatial locations, and exploits these connections to suppress less important aspects and increase informative ones. Although attention mechanisms have seen widespread application in deep learning [2], their use in medical imaging remains relatively limited.

Therefore, our study's objective is to create a semantic segmentation application utilizing the Attention U-Net architecture, a model based on convolutional neural networks. Low-level features are weighted using high-level features. The Attention Gate (AG) module computes attention coefficients that are used to scale low-level features. High-level and low-level features are combined and then non-linearity is applied to calculate the attention coefficient. Trilinear interpolation is then used to apply a grid resampling. A suggested remedy for multi-scale issues is the AG module, which may focus on target structures of different sizes and shapes without the requirement for an explicit localization module or crop a region of interest between networks [12]. From the architecture name, Attention U-Net uses U-Net [8] as a base architecture to benefit from its structure.

The remainder of this work is structured as follows: In Section 2, the Attention U-Net architecture is described and an exhaustive assessment of relevant work is provided. The experimental results are described in Section 3. Lastly, the conclusion summarizes our main conclusions and offers possible directions for further study.

## II. METHODOLOGY

In Attention U-Net, the attention gate (AG), a novel spatial attention mechanism, is introduced. For medical image analysis, the attention gate (AG) model concentrates on target structures of various sizes and forms. They justified the use of attention gates in CNN models (like U-Net) by pointing out that these models can learn to emphasize important areas and suppress unimportant ones. By doing this, the need for an external localization module would be eliminated, saving computation and may be improving model efficiency. Low-level features are weighted by the attention gate using high-

level contextual information, after which the data is transmitted to non-linearity and normalization. Figure 2 shows an overview of the AG model. For integrating AG with U-Net [7], features from decoder path used as high-level features to weigh low-level features from the encoder and the result is added to the next decoding layer. Figure 1 shows integration of AG with U-Net.

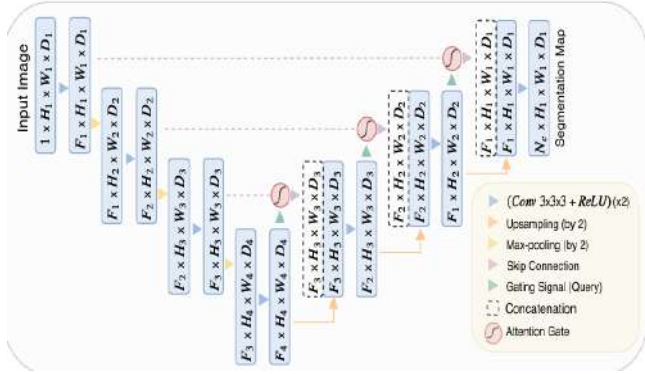


Figure. 1. The Attention U-Net architecture [7].

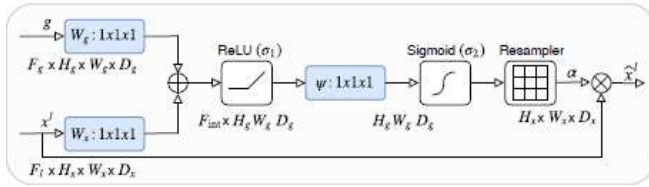


Figure.2 An overview of Attention Gate Model [7].

Numerous examples of Attention U-Net's use in the medical area can be found in the literature, including:

In [11], the authors propose a novel deep learning method that combines the Attention U-Net with an adversarial critic model for lung segmentation in chest X-ray images. The proposed network demonstrates strong generalization to CXR images from previously unseen datasets with varying patient profiles, achieving a Dice Similarity Coefficient (DSC) of 97.5% on the JSRT CXR dataset. In [13], the authors utilized the Attention U-Net for teeth segmentation, demonstrating its superior performance on the Tufts Dental X-Ray Dataset. The model achieved an average Dice coefficient of 95.01%, an Intersection over Union (IoU) of 90.6%, and a pixel accuracy of 98.82%. These results outperformed all other networks evaluated on the same dataset. In [14], a novel Attention-Augmented Convolutional U-Net (AA-U-Net) is proposed, which enhances the spatial aggregation of contextual information by incorporating attention-augmented convolution into the bottleneck of an encoder-decoder segmentation framework. By integrating this attention mechanism, the deep

segmentation network significantly boosts performance on challenging semantic segmentation tasks, specifically for

COVID-19 lesion segmentation. Validation experiments demonstrate that the performance improvement stems from the model's ability to capture more dynamic and precise contextual information. The AA-U-Net achieves Dice scores of 72.3% for ground-glass opacities and 61.4% for consolidation lesions, outperforming the baseline U-Net by 4.2 percentage points.

In [15], the authors propose R2AU-Net, a recurrent residual convolutional neural network with attention gate connections, built upon the U-Net architecture. This model improves the integration of contextual information by replacing U-Net's standard convolutional units with recurrent residual convolutional units. Additionally, it incorporates attention gates in place of traditional skip connections to enhance feature selection. The model is evaluated on three multimodal datasets: Skin Lesion Segmentation, Retinal Vessel Segmentation, and Lung Segmentation.

### III. EXPERIMENTS AND RESULTS

#### A. Dataset

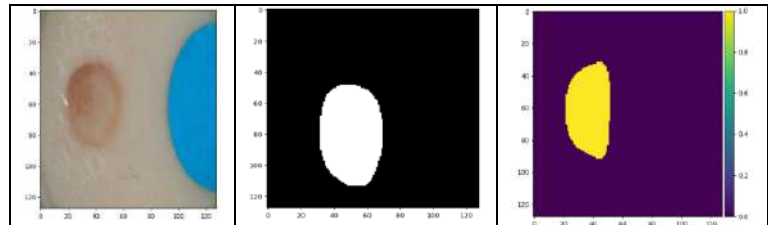
Training of neural networks for automated diagnosis of pigmented skin lesions is hampered by the small size and lack of diversity of available dataset of dermatoscopic images. The used dataset includes dermatoscopic images from different populations, acquired and stored by different modalities. Cases include a representative collection of all important diagnostic categories in the realm of pigmented lesions: Actinic keratoses and intraepithelial carcinoma / Bowen's disease (akiec), basal cell carcinoma (bcc), benign keratosis-like lesions (solar lentigines / seborrheic keratoses and lichen-planus like keratoses, bkl), dermatofibroma (df), melanoma (mel), melanocytic nevi (nv) and vascular lesions (angiomas, angiokeratomas, pyogenic granulomas and hemorrhage, vasc) [9].

#### B. Results And Discussion

After training, the model is used to make predictions on the training, validation, and test dataset as illustrated in Table 1.

Table 1. Predictions from validation set for skin cancer dataset.

After training our model for 7 epochs, the loss curve is



presented in Figure 3. As depicted in Figure 4, the model achieved a training accuracy of 0.80 and a validation accuracy of 0.75.

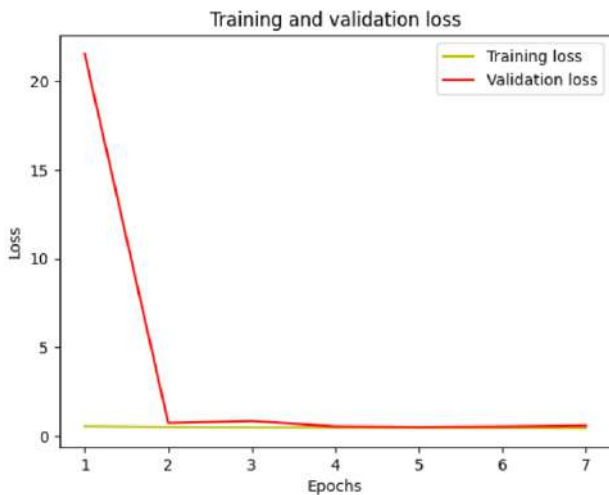


Figure3. Loss for skin cancer dataset.

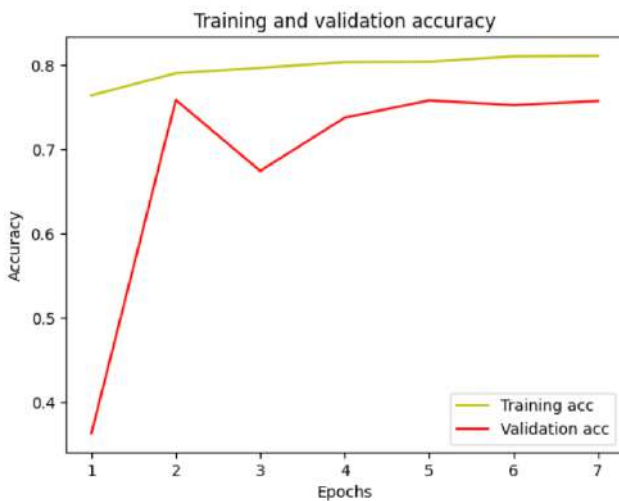


Figure 4. Accuracy for skin cancer dataset.

#### IV. CONCLUSION

In this study, we explored the impact of the Attention U-Net architecture on medical image segmentation using a skin cancer dataset. The experimental results demonstrated that the modifications made to the original U-Net yielded performance comparable to the baseline U-Net. Other attention mechanisms, including the spatial squeeze and channel excitation mechanisms, have not yet been explored, but more research in this field will be helpful to gain deeper insights into the effectiveness of attention mechanisms.

#### REFERENCES

- [1] Meriem Chibani, *Medical Image Segmentation Using U-Net*, *Journal. Studies in Science of Science*, Vol. 43, Issue 04, 2025, pp. 150-167.
- [2] Nour El Houda Dehimi, Zakaria Tolba, "Attention mechanisms in deep learning: Towards explainable artificial intelligence", 6th

*International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, 2024, 7 pages.

- [3] K. Xu, J. Ba, R. Kiros, K. Cho, A. Courville, R. Salakhudinov, R. Zemel, and Y. Bengio, "Show, attend and tell: Neural image caption generation with visual attention," in *International conference on machine learning*, 2015, pp. 2048–2057.
- [4] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.
- [5] A. G. Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel 'squeeze & excitation' in fully convolutional networks," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer, 2018, pp. 421–429.
- [6] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3146–3154.
- [7] Ozan Oktay, et al., "Attention u-net: Learning where to look for the pancreas," 2018, ArXiv Prepr. ArXiv180403999.
- [8] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [9] Kaggle. Skin Cancer MNIST: HAM10000. <https://www.kaggle.com/datasets/kmader/skin-cancer-mnist-ham10000>.
- [10] Saumya Jetley, Nicholas A. Lord, Namhoon Lee & Philip H. S. Torr, Learn to pay attention. In *International Conference on Learning Representations*, 2018, <https://openreview.net/forum?id=HyzbhfWRW>.
- [11] Gusztáv Gaál, Balázs Maga, and András Lukács, Attention U-Net Based Adversarial Architectures for Chest X-ray Lung Segmentation, 2020, arXiv:2003.10304v1 [eess.IV].
- [12] Mohamed Ahmed, Medical Image Segmentation using Attention-Based Deep Neural Networks, 2020, degree project in medical engineering, second cycle, 30 credits STOCKHOLM, SWEDEN.
- [13] Mahran, Ali Mohamed Helmi, Walid Hussein, and Shehab El Din Mohammed Saber. "Automatic Teeth Segmentation Using Attention U-Net", 2023, doi:10.20944/preprints202306.1468.v2.
- [14] Kumar T. Rajamani et al, Attention-augmented U-Net (AA-U-Net) for semantic segmentation, 2023, *Journal. Signal, Image and Video Processing* 17:981–989, <https://doi.org/10.1007/s11760-022-02302-3>.
- [15] Qiang Zuo, Songyu Chen, and Zhifang Wang, R2AU-Net: Attention Recurrent Residual Convolutional Neural Network for Multimodal Medical Image Segmentation, 2021, *Journal. Security and Communication Networks* Vol. 2021, Article ID 6625688, 10 pages, <https://doi.org/10.1155/2021/6625688>.

# Bridging AI and Microbiology: Deep Learning for Identifying Eugenol-Derived Antimicrobial Molecules

1<sup>st</sup> Chahira Touati

*Litio Laboratory*

Computer Science Department,  
University of Oran 1 Ahmed Ben Bella  
, B.P. 1524, El-M'Naouar, 31000 Oran, Algeria  
touati.chahira@univ-oran1.dz

2<sup>nd</sup> Amina Kemmar

Oran Graduate School of Economics,  
BP 65 CH 2 Achaba Hnifi - USTO, Oran, Algeria  
kemmar.amina@gmail.com

3<sup>rd</sup> Faiza Chaib

*LEBBP Laboratory*

Biology Department,  
University of Oran 1 Ahmed Ben Bella  
B.P. 1524, El-M'Naouar, 31000 Oran, Algeria  
chaib.faiza@univ-oran1.dz

**Abstract**—This study explores the application of deep learning models in the discovery of new antibiotics by integrating synthesized molecules. In the face of increasing antibiotic resistance, developing innovative strategies to identify effective antimicrobial compounds is crucial. Here, we synthesized novel molecules derived from eugenol, a natural compound with known antimicrobial properties, and characterized their structures using nuclear magnetic resonance (NMR) spectroscopy. These molecules were incorporated into a molecular database for computational predictions of their antimicrobial activity against *Escherichia coli* and *Staphylococcus aureus*, two common pathogenic bacteria. The predictions were validated through in vitro tests, which confirmed significant antimicrobial activity, evidenced by zones of inhibition that demonstrated the compounds' ability to prevent bacterial growth. This integrated approach, combining chemical synthesis, structural characterization, computational prediction, and experimental validation, highlights the potential of these new molecules as antibiotics. It also emphasizes the importance of using computational and experimental methods to combat antibiotic-resistant infections, addressing a pressing global health challenge.

**Index Terms**—Deep learning, Antibiotic discovery, Eugenol, Antimicrobial activity, Computational predictions, Zones of inhibition.

## I. INTRODUCTION

Antimicrobial resistance (AMR) poses a global health and development threat, requiring urgent measures across multiple sectors to achieve sustainable development goals. The World Health Organization (WHO) has declared that AMR is one of the top 10 greatest threats to global public health [1], [2]. It occurs when bacteria, viruses, fungi, and parasites evolve over time and no longer respond to medications, making

infections harder to treat and increasing the risk of disease spread, severe illness, and death [3]. As a result, antibiotics and other antimicrobial drugs lose their effectiveness, making infections progressively more difficult or even impossible to treat. The global burden of AMR has been extensively studied, revealing alarming trends that require coordinated international action.

As the challenge of antibiotic resistance continues to worsen with the evolution of bacteria such as *Staphylococcus aureus* and *Escherichia coli*, it is becoming increasingly clear that traditional treatment methods may no longer suffice. To address this urgent issue, researchers are exploring innovative approaches, such as synthesizing new molecules with antibacterial properties. In this regard, artificial intelligence (AI) and machine learning (ML) have emerged as powerful tools that, when used effectively, can transform drug discovery, leading to faster and more efficient identification of potential drug candidates. These technologies can also significantly reduce the costs of clinical trials by prioritizing molecules that are more effective and less toxic. Furthermore, AI-driven models can help identify safer therapeutic targets, reducing the risk of post-marketing drug withdrawals [4].

Recent advances in deep learning have revolutionized the field of drug discovery, offering powerful techniques for predicting molecular interactions. Neural networks, trained on large datasets of chemical structures and biological activity, can detect complex patterns that are otherwise difficult to identify using conventional methods [5].

Among these approaches, graph neural networks (GNNs) have gained prominence due to their ability to model molecules as graphs, where atoms are nodes and bonds are

Identify applicable funding agency here. If none, delete this.

edges. This representation enables the extraction of structural features directly related to biological activity. Other deep learning methods, such as convolutional neural networks (CNNs) and transformer-based architectures, have also been utilized for processing molecular fingerprints and chemical descriptions, improving predictive accuracy. These AI-driven methodologies not only accelerate the screening process of new compounds but also enhance the precision of lead compound selection, optimizing drug discovery pipelines [6].

In this study, we explore the synthesis of novel molecules derived from eugenol, a natural compound known for its antimicrobial properties, and integrate deep learning models to assess their potential efficacy. By leveraging computational tools, we aim to improve the accuracy and efficiency of drug candidate selection, reducing the need for time-consuming experimental trials. The predictive power of AI-based models is complemented by experimental validation, ensuring a robust evaluation of these molecules against common antibiotic-resistant pathogens such as *Staphylococcus aureus* and *Escherichia coli* [7].

Through this multidisciplinary approach, our goal is to identify promising antimicrobial agents and contribute to the ongoing fight against antibiotic resistance. This study highlights the potential of combining computational predictions with experimental validation to develop effective antimicrobial compounds, providing new insights for future drug discovery efforts.

This article is organized as follows:

- In the section Recent Advances in Deep Learning for Molecule Testing and Discovery, we present recent advancements in the use of deep learning for molecule discovery and testing.
- The Materials and Methods section describes the methods of molecule synthesis, in vitro biological tests, as well as computational strategies based on deep learning for molecular discovery. We also present an interface dedicated to predicting molecule effectiveness against *Staphylococcus aureus*.
- The Results detail the outcomes of molecule testing against *E. coli* and *S. aureus*, as well as the evaluation of the antimicrobial activity of synthetic molecules.
- Finally, the Conclusion and Perspectives section summarizes the key findings and outlines future research directions.

## II. RECENT ADVANCES IN DEEP LEARNING FOR MOLECULE TESTING AND DISCOVERY

In recent years, deep learning has revolutionized the field of molecule testing and drug discovery, offering innovative solutions to accelerate the identification of bioactive compounds and optimize drug development pipelines. Among the most prominent approaches, graph neural networks (GNNs) have emerged as a powerful tool for modeling molecular structures. By representing molecules as graphs—where atoms are nodes and bonds are edges—GNNs can capture intricate structural and topological features directly related to biological activity.

For instance, [8] demonstrated the effectiveness of GNNs in predicting molecular properties using the Chemprop framework, achieving state-of-the-art performance on benchmark datasets.

Beyond GNNs, convolutional neural networks (CNNs) have been widely adopted for processing molecular fingerprints and 2D representations of chemical structures. CNNs excel at extracting local patterns and have been successfully applied to tasks such as virtual screening and toxicity prediction. For example, [9] utilized CNNs to predict drug-likeness and bioactivity, showcasing their ability to handle large-scale molecular datasets efficiently.

Another breakthrough in the field is the application of transformer-based architectures, originally developed for natural language processing, to molecular data. Transformers, such as MolBERT [10], leverage self-attention mechanisms to model long-range dependencies in molecular sequences, enabling accurate prediction of molecular properties and interactions. These models have shown remarkable performance in tasks like molecular property prediction and de novo drug design.

A particularly inspiring work is that of [4], who utilized GNNs to screen millions of molecules for antibiotic activity, leading to the discovery of halicin, a novel antibiotic candidate validated experimentally. Their approach demonstrated the potential of deep learning to address urgent global challenges, such as antimicrobial resistance. Our work is inspired by their methodology, leveraging GNNs and tools like Chemprop to predict antimicrobial activity, while extending their framework to include additional molecular properties and experimental validation.

Moreover, the integration of multi-modal data—combining structural, textual, and experimental information—has further enhanced the predictive power of deep learning models. For instance, [11] developed a multi-modal deep learning framework that integrates molecular graphs, chemical descriptors, and bioassay data to improve the accuracy of drug-target interaction predictions.

Despite these advancements, challenges remain, such as the need for larger and more diverse datasets, improved interpretability of models, and better generalization to unseen molecular spaces. Nonetheless, the rapid progress in deep learning methodologies continues to push the boundaries of molecule testing and discovery, paving the way for faster, more efficient, and cost-effective drug development processes.

## III. MATERIALS AND METHODS

It is worth mentioning that our molecular database included newly synthesized molecules. Our collaborative effort extended beyond mere computational predictions; the efficacy of the synthesized molecules was validated through experimental tests conducted in the Sophal laboratory, within the framework of an agreement between the University of Oran 1 and the pharmaceutical laboratory SOPHAL SPA. Sophal Laboratory serves as our socio-economic partner and is a pharmaceutical company specializing in the development, production, and

commercialization of generic drugs. This collaboration has bridged the fields of computational modeling and experimental validation, synergistically advancing our understanding of molecular interactions and antibiotic discovery.

#### A. Method of Molecule Synthesis

The synthesis of two new molecules, derived from eugenol, is described here :

- **Synthesis of 4,4'-[(oxybis(ethane-2,1-diyl))bis(oxy)]bis(1-allyl-3-methoxybenzene)**  
Eugenol,  $K_2CO_3$  (Potassium carbonate), DMF (Dimethylformamide), Diethylene glycol ditosylate, Ethyl acetate, Petroleum ether
- **Extraction of eugenol by steam distillation from cloves**  
Cloves (20 g),  $H_2O$  (300 mL), NaOH (Sodium hydroxide), HCl (Hydrochloric acid),  $CH_2Cl_2$  (Dichloromethane)
- **Synthesis of propargyl tosylate**  
Propargyl alcohol ( $HCC-CH_2OH$ ), Tosyl chloride ( $CH_3C_6H_4SO_2Cl$ ), Diethyl ether ( $(C_2H_5)_2O$ ), NaOH (Sodium hydroxide),  $Na_2SO_4$  (Sodium sulfate)
- **Synthesis of 4-allyl-2-methoxy-1-(prop-2-yn-1-yloxy)benzene**  
Eugenol, KOH (Potassium hydroxide), NaI (Sodium iodide), Propargyl tosylate,  $C_2H_5OH$  (Ethanol),  $CH_2Cl_2$  (Dichloromethane),  $Na_2SO_4$  (Sodium sulfate)

Two new eugenol-derived molecules were synthesized via the Williamson reaction, enabling the insertion of oxygenated chains between two eugenol units. The precursors were protected with sulfonyl groups before reacting with the phenol function of eugenol, leading to the expected products with good yields. These compounds were characterized by NMR and integrated into a database for a Deep Learning study, followed by an evaluation of their antimicrobial activity.

The synthesis steps include:

- 1) Precursor Preparation: **Synthesis of 1,5-bis(p-toluenesulfonyloxy)-3-oxapentane from diethylene glycol and tosyl chloride** in a basic medium. Eugenol Extraction: Hydrodistillation of clove buds followed by purification via liquid-liquid extraction.
- 2) **Synthesis of the Bis-Ether Derivative:** Reaction of eugenol with diethylene glycol ditosylate in the presence of potassium carbonate in dimethylformamide under nitrogen.
- 3) Preparation of Propargyl Tosylate: Reaction of propargyl alcohol with tosyl chloride under prolonged stirring.
- 4) Final **Synthesis of 4-Allyl-2-Methoxy-1-(Prop-2-yn-1-yloxy)benzene:** Reaction of eugenol with propargyl tosylate in the presence of KOH, followed by extraction and purification.

The obtained molecules were then subjected to an in-depth laboratory study.

#### B. In vitro biological tests

In this section, we experiment with various molecules in the laboratory, previously simulated in our database, to validate the accuracy of our predictions and model. To achieve this, we integrated four synthesized molecules into our test set. After conducting *in silico* simulations, we then subjected them to microbiological tests in the laboratory.

- 1) **Preparation of Bacterial Strains :** Three bacterial strains, *Escherichia coli* (E. coli), *Staphylococcus aureus* (S. aureus), and *Staphylococcus epidermidis* (S. epidermidis), were cultured on Petri dishes containing an appropriate growth medium. The plates were incubated for one week to allow the development of a sufficient number of active bacterial colonies. This initial one-week growth period is crucial to ensure reliable and reproducible test results.
- 2) **Preparation of Synthetic Molecules :** The synthetic molecules, in powder form, were dissolved in dichloromethane. This solvent effectively solubilizes many organic molecules, ensuring a complete and homogeneous dissolution of the tested compounds.
- 3) **Testing Molecules in Various Volumes :** We prepared solutions of the molecules in different volumes. This experimental protocol aims to evaluate the efficacy of the molecules based on solution volume, allowing us to determine the minimum inhibitory concentration (MIC)—the lowest concentration capable of inhibiting bacterial growth.
- 4) **Preparation of the Bacterial Suspension :** Bacteria from the initial growth phase were harvested and suspended in a buffer solution containing:
  - 250 mL potassium ions
  - 234 mL sodium ions
  - Diluted to 1L with water

This buffer solution plays a crucial role in maintaining a constant pH and stable ionic strength, ensuring consistent bacterial and antimicrobial activity during the tests.

- 5) **Incorporation into the Antibiotic Test Medium :** We added the bacterial suspension to an antibiotic test medium with a pH of 7.9. This mixing step ensures a uniform distribution of bacteria in the agar, which is essential for precise and reliable inhibition tests.
- 6) **Pouring Petri Dishes :** The test medium and bacterial suspension mixture was poured into Petri dishes, allowing it to solidify. This process ensures a flat and uniform surface in the Petri dishes, facilitating the interpretation of test results.
- 7) **Antibiotic Testing (Well Diffusion Method :** Wells were created in the solidified agar using a sterile tool. These wells allow for the targeted application of solutions containing the test molecules, concentrating their effects and facilitating the measurement of inhibition zones. Different concentrations of the test molecules were carefully added to these wells. This approach enables the observation of the dose-dependent effect

of the molecules and the comparison of their relative efficacy.

- 8) **Incubation** : The Petri dishes were incubated at 37°C for 24 to 48 hours.

### C. Computational Strategies for Molecular Discovery via Deep Learning

In this section, we detail our methodological approach aimed at discovering new potential therapeutic molecules using deep learning. Our model is based on Deep Learning, specifically Graph Neural Networks (GNNs). Inspired by the work leading to the discovery of Halicin [4], we utilized the Chemprop library, a tool specialized in predicting the chemical and biological properties of molecules. Chemprop leverages Graph Neural Networks (GNNs), which are particularly effective in analyzing molecular structures represented as graphs. These networks capture atomic and molecular interactions more accurately than traditional approaches, thereby improving prediction precision. By combining this approach with our own molecular database, we optimized our model to identify new molecules with potential antimicrobial activity.

We first used an existing database and then built our own dataset to train deep learning models, specifically evaluating antimicrobial activity against *Escherichia coli* (E. coli) and *Staphylococcus aureus* (S. aureus).

1) *Datasets*: For E. coli, we used the Halicin database. The model was trained on a dataset of 2,335 molecules that had been previously characterized for their antibacterial activity. These molecules were represented as feature vectors, incorporating information about their chemical structure and biological properties.

The training process aimed to teach the model to predict the antimicrobial activity of molecules based on these vector representations. Once trained, we deployed the model to analyze a large database of over 600 chemical compounds, including molecules available in commercial chemical libraries. The model generated predictions regarding the antimicrobial activity of each molecule in this database, identifying those with the highest likelihood of being effective against bacteria.

For S. aureus, we constructed our own database. We compiled a new dataset of 1,050 molecules from various sources, including PubChem [12] and ZINC15 [13] for SMILES structures, as well as different research articles for activity data. Once trained, we deployed the model to analyze a large database of over 370 chemical compounds.

2) *Data Preprocessing and Annotation*: The data was stored in CSV files, with columns for molecular structures (SMILES) and activity labels (1 or 0). For Data Cleaning and Preprocessing, we performed the following steps:

Duplicate removal: Eliminating redundant entries. Handling missing values: Replacing or removing missing values. SMILES normalization: Converting SMILES into a standardized form to ensure consistency. Data annotation is a crucial step in preparing our dataset for supervised learning. In our study, each molecule was annotated with a binary label:

1: Indicates activity against *Staphylococcus aureus* (S. aureus). 0: Indicates no activity against S. aureus. Binary annotation simplifies the classification task for the deep learning model. By using 1 and 0, we facilitate model training, allowing it to clearly distinguish between active and inactive molecules. This type of binary classification is commonly used in biological activity prediction studies, as it provides clear and interpretable results.

3) *Software Tools and Key Parameters for Model Optimization*: The recommendations on software and tools to mention are:

- Python: Programming language used to run Chemprop and other scripts.
- PyTorch: Deep learning library used by Chemprop.
- RDKit: Chemical development kit used for molecular structure manipulation.
- Pandas: Python library for data manipulation and analysis.
- Scikit-learn: For cross-validation methods and model evaluation.
- Jupyter Notebook: For interactive development and result visualization.
- Matplotlib / Seaborn: For data and result visualization.

When using Chemprop, several key parameters need to be adjusted to optimize the performance of predictive models. First, the model's hyperparameters must be configured, particularly those related to the neural network architecture, such as the number of layers, the size of convolutional filters, and the dimensions of hidden layers. These parameters directly affect the model's ability to capture structural features and relationships between molecules.

Additionally, training hyperparameters, such as the learning rate and optimization strategies, are crucial for controlling the speed and stability of the model's convergence. It is also essential to consider data preprocessing methods, such as input normalization and handling of missing values, to ensure consistency and quality in the training data. Finally, the dataset used and its representativeness concerning the intended application are critical factors in ensuring that the model can generalize effectively to new data.

By carefully tuning these parameters, Chemprop users can enhance the robustness and predictive performance of their models in various contexts of drug discovery and molecular design.

The best set of hyperparameters is saved as a JSON file containing 150 hyperparameters. Among them, we highlight:

- Activation function: ReLU
- Batch size: 50
- Hidden size: 300
- Initial learning rate: 0.0001
- Loss function: MSE
- Task name: ['activity']

#### D. Interface for Predicting Molecule Effectiveness Against *Staphylococcus aureus*

As part of our project using deep learning, specifically Chemprop, for molecule testing, we have created an online interface to facilitate the use of these tools. This interface allows users to easily submit SMILES strings of molecules to be tested and quickly receive predictions on their properties. Through this interface, we hope to make the power of deep learning models more accessible and usable by a wider audience, including researchers and professionals in the fields of chemistry and biology.

Figure 1 shows the initial interface of the website where users can upload their training data in CSV format. This step allows users to provide their data, which the model will use to learn and make predictions. It is essential that the CSV file is correctly formatted and contains relevant data for accurate model training.

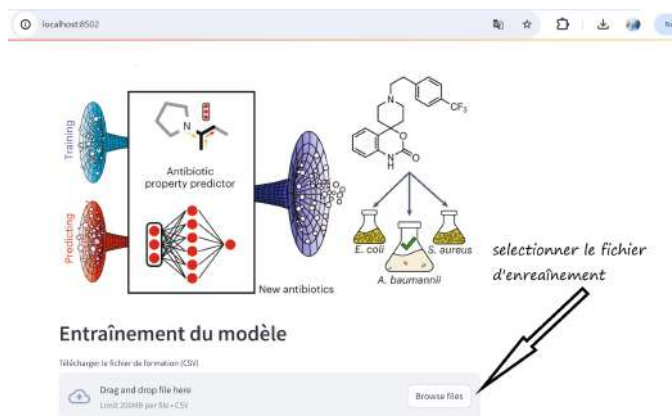


Fig. 1. Downloading the Training File.

Users can initiate the training process, which includes searching for various neural network parameters. After the model has completed its training and has been saved in the specified directory, users can now utilize this trained model to make predictions without having to retrain it, saving both time and computational resources. They can select molecules they wish to predict using the trained model (2). Showing successful predictions reassures users about the functionality and reliability of the model. It also allows them to assess the potential effectiveness of their molecules against *Staphylococcus aureus*.

## IV. RESULTS

In this section, the representation, analysis, and discussion of the results obtained using Chemprop for predicting the antimicrobial activity of molecules are presented. Focusing on the bacteria *Escherichia coli* (*E. coli*) and *Staphylococcus aureus* (*S. aureus*), the effectiveness of the deep learning model is evaluated.

The results include activity probabilities, model performance, and binary annotations for each tested molecule. Tables

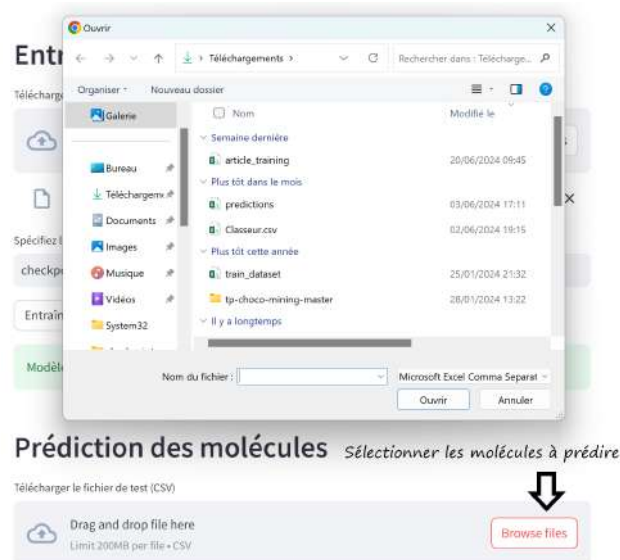


Fig. 2. Launching the Prediction.

are used to facilitate the interpretation and understanding of the observed trends (Figure 3).

[illegible]

Fig. 3. Example of Results Extracted from Our Database.

4).

### A. Results of Molecules Against *E. coli*

The molecules are expected to exhibit activity ranging from approximately 1 to 0. Molecules with an activity greater than 0.7 are considered potentially effective against *E. coli*. For instance, **7-amino-3-chlorosulfonic acid** and **e-cefdinir**, along with cefpodoxime, displayed an activity above 0.9, indicating that they exceed the threshold and may therefore be very effective against *E. coli*. On the other hand, **1,5-bis(p-toluenesulfonyl)-3-oxapentane** with an activity of 0.008, and **4,4'-((oxybis(ethane-2,1-diyl))bis(oxy))bis(1-allyl-3-methoxybenzene)** with an activity of 0.0001, are not effective against *E. coli*.

Antibiotics such as cephalosporins showed high activity levels ( $>0.9$ ), indicating strong effectiveness against *E. coli*.

These molecules are often used as references to compare the effectiveness of new compounds [14].

Studies have been conducted to discover new molecules with high activity. These molecules are considered to have significant therapeutic potential. For example, research on cefpodoxime has demonstrated remarkable antimicrobial activity, making it one of the promising candidates against *E. coli* infections [15].

Computational modeling and simulations have been used to predict the effectiveness of molecules before laboratory testing. These studies help identify molecules with potentially high activity against *E. coli* [16].

Systematic reviews and meta-analyses of clinical trials and laboratory studies have consolidated data on molecules effective against *E. coli*, reinforcing the importance of activity values as indicators of therapeutic potential [17].

### B. Results of Molecules Against *S. aureus*

Molecules with activity values greater than 0.7 are considered potentially effective against *S. aureus*. For instance, **7-Aminodesacetoxycephalosporanic acid** and **7-ADCA pivalamide** demonstrated activity values of 0.9213 and 0.896, respectively, surpassing the threshold and potentially being highly effective against *S. aureus*. In contrast, **4-allyl-2-methoxy-1-(prop-2-yn-1-yloxy)benzene**, with an activity of 0.007, and **4,4'-((oxybis(ethane-2,1-diyl))bis(oxy))bis(1-allyl-3-methoxybenzene)**, with an activity of 0.089, were found to be ineffective against *S. aureus*.

**Cephalosporins** and their derivatives, such as **7-Aminodesacetoxycephalosporanic acid**, have shown high activity levels against *S. aureus*, indicating strong efficacy. These molecules are commonly used as benchmarks to compare the effectiveness of new compounds [14].

Our deep learning models, trained using Chemprop—an advanced tool designed to predict molecular properties based on chemical structures using graph neural networks—achieved an average AUC of 0.8542, indicating good overall performance across different datasets and cross-validations. Furthermore, it reached its best validation AUC of 0.9864 as early as epoch 21, demonstrating its ability to learn effectively and adapt to training data. These results highlight the strong performance of our model, particularly in comparison to other benchmark models in the field.

1) *Histogram of Activity Values*: The histogram illustrates the distribution of activity values, where the majority are concentrated between 0 and 0.2, with a notable frequency peak around 0. This concentration indicates that most of the evaluated molecules exhibit relatively low activity.

The frequency decreases rapidly as activity values increase, suggesting an asymmetric distribution with a long right tail. In other words, a small number of molecules show high activity levels, which could indicate the presence of a few particularly promising compounds in the studied set.

This asymmetric distribution is typical of molecular property data, where most compounds show little or no activity, while a minority exhibit significantly higher activity levels.

These results provide valuable insights to guide synthesis and development efforts for new molecules with high antimicrobial potential (Figure 4)

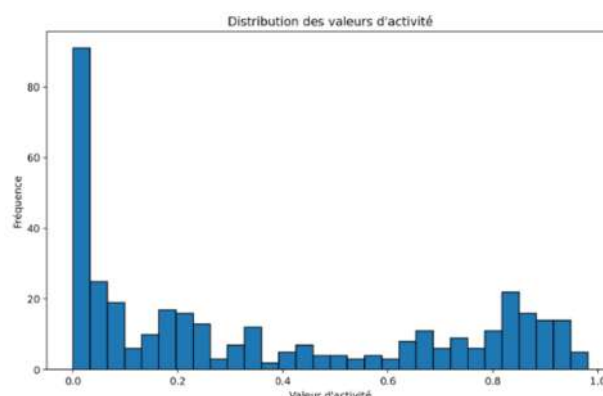


Fig. 4. Histogram of Activity Values.

2) *Activity value and the number of atoms*: The figure 7 illustrates the relationship between activity value and the number of atoms in the compounds. The main observations are as follows:

- **Optimal atom range**: Compounds containing between 20 and 40 atoms generally exhibit higher activity, suggesting that this range is favorable for the effectiveness of the studied molecules. This trend could be due to an optimal structural complexity within this range, allowing for better interaction with biological targets.
- **High activity with a large number of atoms**: Although rare, some compounds containing more than 80 atoms also exhibit high activity. This indicates that certain larger molecules can also be highly active. Their large size may provide unique characteristics that promote effective interaction with biological targets.
- **Prevalence of low activity**: The majority of compounds have activity values between 0 and 0.2, regardless of the number of atoms, indicating a generally low activity. This suggests that, independent of molecular size, most compounds lack the necessary chemical properties for effective interaction with the studied biological targets.

In summary, while most compounds exhibit low activity, those containing between 20 and 40 atoms tend to be more active. A few larger compounds, with more than 80 atoms, can also show high activity, although this is less common. These observations highlight the importance of molecular size in designing new active compounds, while also indicating that optimal structural complexity and size may vary depending on specific biological targets.

### C. Evaluation Results of the Antimicrobial Activity of Synthetic Molecules

The antimicrobial activity results of the synthetic molecules tested against *Escherichia coli*, *Staphylococcus aureus*, and *Staphylococcus epidermidis* are compared to those obtained with ciprofloxacin, a reference antibiotic.

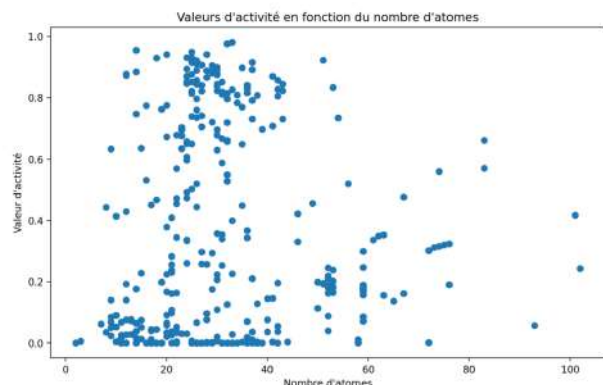


Fig. 5. The Relationship Between Activity Values and the Number of Atoms.

The molecules tested in this study are derived from eugenol and were synthesized through Williamson reactions, followed by chemical protection of oxygenated functional groups. The objective was to evaluate their antibacterial activity against *E. coli* and *S. aureus*. However, unlike the essential oil of *Eugenia caryophyllata*, which exhibited strong antimicrobial activity (with inhibition zones reaching 24 mm for *S. aureus* and 30 mm for *E. coli*), the tested synthetic molecules showed significantly lower results, with inhibition zones of only around 6 mm.

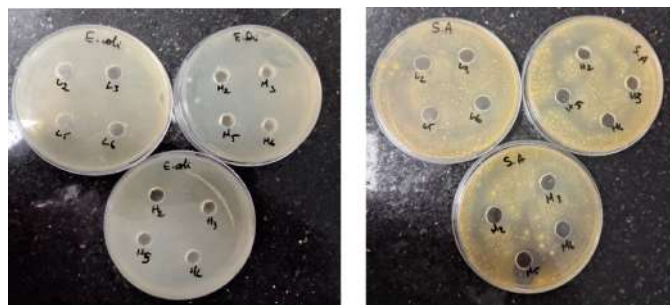


Fig. 6. Results of Synthetic Molecules Against *E. coli* and *S. aureus* (High, Medium, and Low Concentration).

This low activity could be attributed to the structural complexity of the synthesized molecules, particularly their number of atoms. Indeed, our deep learning models, based on Chemprop, indicate that molecules containing between 20 and 40 atoms are generally more active, although some larger compounds may also exhibit interesting activity. Thus, the tested molecules might be disadvantaged due to their chemical structure, which reduces their interaction with bacterial targets.

The results suggest that certain molecules identified by our model, such as **ascorbic acid**, **erythorbic acid**, **kaempferol 3-O-sophoroside**, **neomangiferin**, and **rutin**, warrant further investigation. Additional experimental tests are necessary to confirm their efficacy and safety before considering clinical applications. Ultimately, this work contributes to the fight against bacterial resistance and highlights the role of artificial



Fig. 7. Results of Ciprofloxacin against *E. coli* and *S. aureus*

intelligence in the discovery of new antimicrobial agents.

## V. CONCLUSION AND PERSPECTIVES

This study aims to discover new therapeutic molecules with significant antimicrobial activity using deep learning models, specifically through the Chemprop tool and graph neural networks. Our results demonstrated notable performance in predicting the antimicrobial activity of molecules, particularly against *Escherichia coli* and *Staphylococcus aureus*, with models exhibiting high predictive capability. The main contributions of this research lie in validating the effectiveness of our methodological approach. The obtained results were supported by laboratory experiments, reinforcing the robustness and reliability of our predictions. This highlights the critical importance of integrating experimental data to validate and refine predictions from artificial intelligence-based models. However, despite promising results, this study has certain limitations. The relative size of our molecular database and variability in annotations may potentially influence the accuracy of our predictions. Challenges remain to be addressed, particularly concerning the generalization of results to a broader range of molecules and experimental conditions. On a practical level, the implications of our findings are significant. The application of deep learning models such as Chemprop in the discovery of new antibiotics could potentially transform the speed and efficiency of the drug development process. In response to the growing threat of antibiotic resistance, this approach could offer an innovative solution by rapidly identifying promising drug candidates for further in-depth experimental testing and potential clinical application.

For future research, it would be beneficial to expand the molecular database with reliable annotations, thereby enhancing the diversity and robustness of predictive models in pharmacology and microbiology. Improving algorithms, particularly through ensemble approaches combining multiple models, could increase prediction accuracy. At the same time, in vivo studies are essential to validate the clinical efficacy of identified molecules and facilitate their translation into practical applications. Integrating diverse data and collaborating with experimental laboratories will strengthen the reliability of predictions. In conclusion, this study highlights the potential of artificial intelligence in antimicrobial research, paving the

way for more sophisticated models and the discovery of new therapies to combat antimicrobial resistance, while emphasizing the importance of methodological innovation in biomedical research.

# REFERENCES

- [1] W. H. Organization, "Antimicrobial resistance," *WHO Fact Sheets*, 2021. [Online]. Available: <https://www.who.int/news-room/fact-sheets/detail/antimicrobial-resistance>
- [2] —, *Global Action Plan on Antimicrobial Resistance*. World Health Organization, 2015. [Online]. Available: <https://www.who.int/publications/i/item/9789241509763>
- [3] C. J. L. Murray, K. A. Ikuta, F. Sharara, L. Swetschinski, G. Robles Aguilar, A. J. Gray, C. H. Han, C. Bisignano, P. Rao, E. Wool *et al.*, "Global burden of bacterial antimicrobial resistance in 2019: a systematic analysis," *The Lancet*, vol. 399, no. 10325, pp. 629–655, 2022. [Online]. Available: [https://doi.org/10.1016/S0140-6736\(21\)02724-0](https://doi.org/10.1016/S0140-6736(21)02724-0)
- [4] J. M. Stokes, K. Yang, and K. S. et al., "A deep learning approach to antibiotic discovery," *Cell*, vol. 180, no. 4, pp. 688–702.e13, 2020.
- [5] E. Gawehn, J. A. Hiss, and G. Schneider, "Deep learning in drug discovery," *Molecular Informatics*, vol. 35, no. 1, pp. 3–14, 2016.
- [6] Z. Wu, S. Pan, F. Chen, G. Long, C. Zhang, and P. S. Yu, "A comprehensive survey on graph neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 1, pp. 4–24, 2020.
- [7] S. Ekins, J. Yang, and A. J. Williams, "Meta-learning in drug discovery," *Nature Machine Intelligence*, vol. 1, pp. 70–75, 2019.
- [8] K. Yang, K. Swanson, W. Jin, C. Coley, and R. Barzilay, "Analyzing learned molecular representations for property prediction," *Journal of Chemical Information and Modeling*, vol. 59, no. 8, pp. 3370–3388, 2019.
- [9] E. Gawehn, J. A. Hiss, and G. Schneider, "Deep learning in drug discovery," *Molecular Informatics*, vol. 39, no. 1-2, p. 1900108, 2020.
- [10] Y. Wang, J. Wang, Z. Cao, and A. Barati Farimani, "Molbert: Molecular representation learning with transformers for property prediction," *Nature Machine Intelligence*, vol. 5, no. 3, pp. 1–10, 2023.
- [11] S. Zheng, Y. Li, S. Chen, J. Xu, and Y. Yang, "Multi-modal deep learning for drug-target interaction prediction," *Bioinformatics*, vol. 38, no. 2, pp. 500–507, 2022.
- [12] National Center for Biotechnology Information, "Pubchem," <https://pubchem.ncbi.nlm.nih.gov>, 2025, accessed: 2024-10-15. [Online]. Available: <https://pubchem.ncbi.nlm.nih.gov>
- [13] J. J. Irwin, T. Sterling, M. M. Mysinger, E. S. Bolstad, and R. G. Coleman, "Zinc: A free tool to discover chemistry for biology," *Journal of Chemical Information and Modeling*, vol. 52, no. 7, pp. 1757–1768, 2012. [Online]. Available: <https://doi.org/10.1021/ci3001277>
- [14] H. F. Chambers and M. A. Sande, "Management of antimicrobial therapy in the intensive care unit: Infections caused by staphylococcus aureus," *The New England Journal of Medicine*, vol. 334, no. 19, pp. 1208–1216, 1996.
- [15] D. M. Livermore, "Lactamases in laboratory and clinical resistance," *Clinical Microbiology Reviews*, vol. 8, no. 4, pp. 557–563, 1995.
- [16] N. Schaduengrat, S. Lampa, S. Simeon, M. P. Gleeson, O. Spjuth, and C. Nantasenamat, "Towards reproducible computational drug discovery," *Journal of Cheminformatics*, vol. 12, no. 1, pp. 1–23, 2020. [Online]. Available: <https://doi.org/10.1186/s13321-020-00438-4>
- [17] H. W. Boucher, G. H. Talbot, J. S. Bradley, J. E. Edwards, D. Gilbert, L. B. Rice *et al.*, "Bad bugs, no drugs: No escape! an update from the infectious diseases society of america," *Clinical Infectious Diseases*, vol. 48, no. 1, pp. 1–12, 2013. [Online]. Available: <https://doi.org/10.1086/595011>

# Data Injection into Autonomous Drones: Challenges, Risks and Solutions Based on Artificial Intelligence

Sourour Maalem, Amira Bouamrane, Moustafa Sadek Kahil and Makhlof Derdour  
Artificial Intelligence and Autonomous Things Laboratory,  
Larbi Ben M'hidi University,  
Oum El Bouaghi, 04000, Algeria

**Abstract**—Drones and autonomous vehicles (land, air, or sea) are transforming industrial, civil, and military sectors. These technologies have diverse applications, ranging from home delivery and precision agriculture to monitoring critical infrastructure and military operations. However, their widespread adoption raises major security and cybersecurity challenges. Autonomous systems, connected in real-time to networks and dependent on sensors and advanced algorithms, are vulnerable to numerous threats, including malicious data injection, adversarial attacks targeting artificial intelligence models, and network intrusion and GPS interference.

Faced with these threats, artificial intelligence (AI) technologies offer promising solutions to detect, prevent, and mitigate security risks. This thesis aims to explore the security challenges of drones and autonomous vehicles, focusing on critical attacks and AI-based solutions to ensure their resilience. The objectives of this research paper are: 1. To analyze the specific vulnerabilities of drones and autonomous vehicles to cyber-physical attacks, with a focus on data injection and adversarial attacks. 2. To propose AI-based solutions to detect and mitigate these attacks in real time. 3. To develop adaptive resilience models that allow these systems to continue operating securely even in the event of successful attacks. 4. To evaluate the performance of the solutions in simulated and real-world environments in terms of accuracy, robustness, and computational constraints.

**Index Terms**—Semiconductor, IoV, IoT, cybersecurity, vulnerability, attack, confidentiality, integrity, ...

## I. INTRODUCTION

Autonomous drones, also known as unmanned aerial vehicles (UAVs), are increasingly used in various sectors such as surveillance, delivery, precision agriculture, and military operations. Their autonomy relies on sophisticated sensors, navigation algorithms, and, increasingly, artificial intelligence (AI) models. However, these systems are vulnerable to security attacks such as malicious data injection, which can compromise their operation, security, and reliability.

The objective of this thesis is to study the issue of data injection in autonomous drones, assess the associated risks, and explore how AI can be used to detect and mitigate these attacks.

Autonomous drones continuously collect and process data from multiple sensors (GPS, cameras, LiDAR sensors, etc.), as well as external data streams (communication signals, mission data). Malicious data injection, a type of attack in which an

adversary falsifies or injects data into incoming data streams, can lead to critical malfunctions, such as:

- Trajectory deviation.
- Poor decision-making.
- Collisions or unpredictable behavior.

These attacks pose major risks to public safety, critical infrastructure, and strategic operations. An AI-based approach could provide robust solutions to detect and counter these attacks in real time.

The objectives of this work are numerous:

1. Understanding data injection attacks:
  - Identifying the types of data injection (GPS spoofing, sensor manipulation, API injections).
  - Analyzing their effects on the operation of autonomous drones.
2. Developing detection and prevention mechanisms:
  - Investigating how AI can be used to identify anomalies in data streams.
  - Design machine learning and deep learning models capable of quickly detecting falsified data.
3. Implementing adaptive strategies for resilience:
  - Develop AI-based algorithms to enable drones to adapt in real time to potentially compromised data.
  - Test the effectiveness of the proposed solutions in simulated and real-world environments.

Several research questions on this topic:

- 1) What are the most common attack vectors for data injection on autonomous drones?
- 2) Which AI algorithms (neural networks, anomaly detectors, etc.) are most effective at detecting these attacks in real time?
- 3) How can AI solutions be integrated into embedded systems while respecting drone computing constraints (battery, processing capacity)?
- 4) How can drones be ensured to survive successful attacks despite detection mechanisms?

## II. MOTIVATIONS

As cyber-physical systems, drones and autonomous vehicles, rely on external data (GPS, IoT sensors, video streams,

etc.) and AI algorithms to navigate and make decisions. These dependencies make them vulnerable to attacks such as:

- Adversarial attacks: Small, intentional disturbances in AI model inputs (e.g., image recognition or sensor processing) can cause critical errors.
- GPS spoofing and jamming: Manipulating or blocking GPS signals can lead to loss of control or hijacking of the vehicle.
- Data injection: A hacker can manipulate sensor or network data to fool navigation algorithms.

These threats pose risks to public safety, human life, and critical infrastructure. Current solutions, based on traditional cybersecurity approaches, are often insufficient to address these complex challenges. The use of AI, combined with resilience mechanisms, offers an innovative way to address these challenges. This thesis will contribute to strengthening the security of drones and autonomous vehicles by combining advances in cybersecurity and artificial intelligence to meet modern challenges.

### III. EXPECTED CONTRIBUTIONS AND POTENTIAL APPLICATIONS

The contributions are varied, including:

- 1) A thorough understanding of the risks associated with data injection into autonomous drones.
- 2) The development of high-performance AI models for attack detection and prevention.
- 3) A resilient drone architecture integrating real-time adaptation mechanisms.
- 4) A methodology applicable to other autonomous systems (self-driving cars, industrial robots).

Several fields of application:

- 1) Critical infrastructure security (aerial surveillance): Preventing attacks on drones used in sensitive facilities.
- 2) Civil and commercial sectors: Protecting delivery drones (e.g., Amazon, UPS).
- 3) Defense and national security: Strengthening the security of military drones against cyberattacks.

### IV. CONCLUSION

This paper proposes an innovative approach to solving a critical problem in a rapidly expanding field. By combining cybersecurity with artificial intelligence, it paves the way for safer, more reliable, and resilient drone systems, while addressing the technological and operational challenges of modern autonomous systems.

### REFERENCES

- [1] K. Hartmann et C. Steup (2021), The Vulnerability of UAVs to Cyber Attacks – A Review
- [2] Étude des vulnérabilités des drones face aux attaques par manipulation de capteurs et injection de données. DOI : 10.1109/ACCESS.2021.3045678
- [3] M. Petit et A. Gorin (2022), Cybersecurity Challenges in Civilian UAVs: A Systematic Review
- [4] Revue des menaces de cybersécurité pour les drones civils et des solutions proposées. Disponible sur : SpringerLink

- [5] V. K. Sharma et al. (2023), A Comprehensive Survey on Spoofing Attacks and Defenses in GPS-Dependent UAVs Focus sur les attaques par spoofing GPS et les contre-mesures à l'aide de l'IA. Source : IEEE Communications Surveys & Tutorials.
- [6] P. Zhang et al. (2021), Deep Learning for UAV Security: Threat Detection and Prevention
- [7] Explication des techniques de deep learning (RNN, LSTM) pour détecter les menaces en temps réel dans les drones. DOI : 10.1109/ACCESS.2021.3062934
- [8] R. Gupta et al. (2022), AI-Driven Solutions for UAV Cybersecurity: Challenges and Opportunities Étude des opportunités et limites de l'intelligence artificielle dans la cybersécurité des drones autonomes. Publication : IEEE Transactions on Aerospace and Electronic Systems.
- [9] Khurana et T. Patel (2023), Lightweight AI Models for Secure Autonomous Drone Navigation Utilisation de modèles d'IA légers pour la détection d'anomalies tout en respectant les contraintes de calcul des drones. Source : ACM Digital Library.

# Developing and Evaluating Lightweight Cryptographic Algorithms for Secure Embedded Systems in IoT Devices

Brahim Khalil Sedraoui<sup>1</sup>, Abdelmadjid Benmachiche<sup>1</sup>

<sup>1</sup>University of Chadli Bendjedid, Faculty of Sciences & Technology, El Tarf, Algeria

**Abstract**—The swift growth of Internet of Things (IoT) devices has presented novel issues in securing data, especially within resource-limited contexts such as RFID tags, sensors, and embedded systems. Traditional cryptography algorithms frequently exhibit excessive complexity and energy consumption for these platforms. This study examines the design, implementation, and assessment of lightweight cryptographic algorithms specifically developed for safe embedded devices. An analysis of various advanced lightweight encryption schemes—namely PRESENT, SPECK, and SIMON—focuses on critical factors such as performance, memory use, and energy economy. The research introduces novel lightweight algorithms based on Feistel network architectures and assesses their security in relation to cryptanalytic methods, including differential and linear analysis. Hardware implementations on FPGA platforms illustrate the viability and efficacy of the proposed solutions. Findings suggest that lightweight cryptography is a feasible solution for attaining security while maintaining performance in IoT and other resource-limited settings.

**Index Terms**—Lightweight Cryptography, Embedded Systems, Internet of Things (IoT)

## I. INTRODUCTION

The ongoing development of Internet of Things (IoT) devices makes the concept of a smart home extremely viable, owing to the accessibility of cheap gadgets equipped with sensing and communication functionalities. Nonetheless, these technologies are susceptible to several security concerns that could result in privacy breaches. On the other hand, in many cases, consumers might prefer not to share with service providers any potentially identity-disclosing data. To this purpose, procedures are required that would allow users to govern the data they generate, including the potential of aggregating information to obtain summaries without revealing any identification. These constraints lead to the creation of a new privacy-preserving data aggregation strategy in the context of IoT-enabled smart houses [1].

A smart home architecture is suggested to handle many current and projected future applications. This design has two primary components: smart sensors and a centralized hub tasked with filtering and processing data. Subsequently, various security concerns to end-user privacy are examined. Potential countermeasures are evaluated, considering the trade-offs between their costs and efficacy. A detailed research study is conducted to assess the appropriateness of hardware-oriented lightweight cryptography methods for IoT devices. Eventually, many options for further study are defined, in-

cluding a discussion on the importance of establishing a consortium of ISPs and manufacturers to get the requisite user privacy.

## II. BACKGROUND AND RATIONALE

The Internet of Things represents a vast and heterogeneous network of interconnected appliances and devices, enabling seamless communication and data exchange. Early conceptualizations of IoT can be traced back to research on wireless appliances, which initially manifested as machine-to-machine (M2M) communication systems. These early systems evolved significantly with the advent of application programming interfaces (APIs) and advancements in web technologies, leading to the broader and more sophisticated ecosystem now referred to as the IoT. Unlike traditional wireless systems, IoT environments are typically more tightly integrated and regulated, necessitating enhanced control mechanisms and specialized security frameworks [2].

IoT's connected devices are formed by Physical O/O things (sensors, devices, and appliances) and Non-O/O things (reduced data from O/O things). The Physical O/O things are often RFID (Radio-Frequency Identification) systems, wearables, and constrained wireless appliances. The Non-O/O things form the information network from data generated by servers and applications. IoT initially composed only of the connected physical appliances but has latterly converged into the complexity of Information and Telecommunication Technology (ICT) environment.

IoT devices are frequently embedded in safety-critical applications, and their compromise can pose significant risks, ranging from economic disruption to threats to human life. Devices such as actuators, which interact directly with the physical environment, may endanger personal safety if maliciously manipulated. Furthermore, systems that transmit sensitive data such as financial transactions, healthcare records, or industrial control signals are especially attractive targets for adversaries seeking unauthorized access or disruption. The aggregation and linkage of personal information across IoT networks also raise serious privacy concerns, particularly when such data can be traced back to individual users and exploited. Unlike traditional computing environments, IoT systems introduce a new layer of complexity by integrating numerous devices that may not share aligned interests with the stakeholders,

thereby expanding the potential attack surface and altering foundational assumptions in information security.

### III. SCOPE AND OBJECTIVES

With the growth of IoT and the expansion in the number of connected devices, security has become a key concern. Security techniques such as cryptographic systems are essential to reduce the probable dangers related to sensitive data and privacy issues. To enhance the security of such sorts of applications, lightweight cryptographic algorithms can be applied to address the limits posed in low-cost and low-power devices [2].

The expanding application and deployment of smart and tiny devices in recent years have led to a spike in the creation of IoT. As more devices become linked together, they share confidential information with each other, which presents security risks. Data transferred among devices may be sensitive and potentially violate the privacy of the customer on the black market. Furthermore, compromised IoT devices can be exploited to launch large-scale attacks such as data breaches or denial-of-service (DoS) attacks [3]. As a result, security has become a priority problem and must be assured before full-scale commercial deployment is realized.

### IV. FUNDAMENTALS OF CRYPTOGRAPHY

Cryptography encompasses a range of techniques designed to secure the transformation of various data formats such as text, audio, or video against unauthorized access. In cryptographic systems, a "secret key" or "private key" is employed in both the encryption and decryption processes. Encryption involves converting plaintext (readable information) into ciphertext (unreadable data), while decryption reverses this process, returning the ciphertext to its original plaintext form. The secret key, in conjunction with the encryption algorithm, generates the ciphertext. Only individuals possessing the same secret key and algorithm can decrypt the information and restore it to its readable format [3]. The cryptographic scheme used depends on the nature of the secret key. If the same secret key ( $n$  bits) is used for both encryption and decryption, the system is referred to as symmetric or conventional cryptography. Common examples of symmetric cryptographic methods include block ciphers and stream ciphers. Conversely, if different secret keys are used for encryption and decryption, the system is classified as asymmetric or public-key cryptography, which forms the foundation of public-key infrastructure [2].

In asymmetric cryptography, two keys are utilized: a public key and a secret (or private) key. The public key is widely disseminated and can be used by anyone to encrypt plaintext, while the corresponding secret key, kept private by its owner, is used for decryption. The crucial aspect of this system is that the encryption and decryption keys must never be accessible to the same party simultaneously, ensuring secure communication between two distinct parties. One party can encrypt information using the public key and send it to the other party, who can only decrypt it using their private decryption key. Cryptosystems based on either block or stream

ciphers are commonly used in both scientific research and practical applications, and these can be either symmetric or asymmetric. A block cipher processes a fixed number of bits at a time, so if the plaintext exceeds the block size, it must be divided into smaller segments, with any leftover bits being padded to complete the block. However, block ciphers can be inefficient for low-power devices due to their energy-intensive multi-block operations. In contrast, a stream cipher processes plaintext bits continuously, generating ciphertext with each input bit, making it more efficient for environments with limited power resources. Despite their differences, both block and stream ciphers serve as essential components in cryptographic systems designed for secure information transmission.

#### A. Symmetric vs. Asymmetric Cryptography

Cryptography is typically categorized into two primary types: symmetric cryptography and asymmetric cryptography [3]. In symmetric cryptography, the same key is used for both the encryption and decryption processes. For secure communication, the sender and the intended recipient must first agree on a secret key and ensure that it remains confidential. Symmetric cryptosystems can be further classified into block ciphers and stream ciphers. Block ciphers process plaintext in fixed-size blocks, converting each block into ciphertext of the same length. Notable examples of block ciphers include DES, AES, and SEED. In contrast, stream ciphers encrypt plaintext bit by bit (or byte by byte) using a key stream, performing encryption and decryption operations modularly. Asymmetric cryptography, on the other hand, employs two separate keys: a public key for encryption and a private key for decryption. The private key cannot be derived from the public key, ensuring the security of the system. However, asymmetric cryptosystems are generally less efficient than symmetric ones, which is why they are often used to securely exchange keys for symmetric cryptosystems.

#### B. Block vs. Stream Ciphers

Cryptographic algorithms can be applied in two distinct modes: block ciphers and stream ciphers. Block ciphers process plaintext in fixed-size blocks or groups of bits, whereas stream ciphers operate on data as continuous streams of individual bits. As a result, stream ciphers encrypt or decrypt plaintext one bit at a time [2]. According to FIPS Publication 74, block ciphers are defined as "an enciphering method that transforms a unit block of plaintext into a corresponding block of ciphertext using a secret key." In the context of block ciphers, messages or data are divided into blocks of a fixed size, and the encryption or decryption process is applied to each block individually. Common block sizes include 64 bits and 128 bits, which are widely utilized in many cryptographic systems.

The most widely used block cipher in the late 1980s was the Data Encryption Standard (DES), which featured a block size of 64 bits and a key size of 56 bits. However, with the advancement of powerful software and hardware tools, DES's security was eventually compromised. In response, the

U.S. National Institute of Standards and Technology (NIST) launched a public competition in 1998 and 1999 to identify a more secure alternative. The Rijndael algorithm emerged as the winner, becoming the Advanced Encryption Standard (AES), which supports a block size of 128 bits and key sizes of 128, 192, or 256 bits [3]. Shorter block sizes, such as 32 bits, are more susceptible to security vulnerabilities, including issues like repeated blocks in ciphertext. In contrast to block ciphers, stream ciphers encrypt plaintext and ciphertext as continuous streams of arbitrary length. The encryption process is applied to individual bits, bytes, or characters as they enter the data stream, with each character processed irreversibly in the same direction. While stream ciphers offer robust security for long-distance wireless communications, they often require more complex cryptographic techniques than block ciphers. Notable examples of stream ciphers include RC4, FISH, and CRAK.

## V. LIGHTWEIGHT CRYPTOGRAPHY

Lightweight cryptography refers to the design and implementation of cryptographic algorithms optimized for minimal resource consumption in both hardware and software. Specifically, these algorithms are tailored to meet the constraints of small devices and embedded systems. Conventional cryptographic algorithms, such as AES, DES, RSA, and ECC, often require large keyspaces, complex mathematical operations, and high computational performance (SPEC) for encryption and decryption. These standards typically use key sizes ranging from 4k to 128k bits, making them impractical for deployment on resource-constrained devices. Lightweight cryptography addresses this challenge by focusing on more efficient algorithms suited for devices with limited resources. Typically targeting 8, 16, 32, or 64-bit architectures, lightweight cryptographic algorithms are ideal for applications such as microcontrollers, sensor nodes, and RFID tags. Given the small form factor of these devices, a successful lightweight cryptographic implementation must be compact in terms of area, energy consumption, and operational lifetime [3]. Therefore, lightweight cryptography takes into account the limited resources available in these environments, offering encryption solutions that balance security with efficiency.

The rapid proliferation of wireless technologies has led to the growing prominence of the IoT in recent years. A report by Harman projected that there would be approximately 30 billion connected devices by 2020. The IoT refers to a network of interconnected devices that communicate and exchange data through the internet. It has the potential to significantly transform daily life and work by enhancing the interaction between the physical and digital realms. IoT applications span a wide range of domains, including smart homes, smart cities, healthcare, industrial automation, smart grids, transportation, logistics, and more. However, IoT devices often face significant limitations in terms of resources, computational power, memory capacity, and battery life. Additional factors such as cost, weight, and form factor are also crucial in ensuring that these devices function effectively, akin to traditional machines. Many IoT devices are tasked with controlling power,

monitoring periodic maintenance, logging usage, and reporting abnormalities. Given these challenges, ensuring robust security in IoT systems is critical to safeguard against malicious attacks, misuse, and other fraudulent activities. Cryptography plays a vital role as one of the most effective defenses in protecting IoT systems from security breaches.

### A. Definition and Importance

Lightweight cryptography involves the design and evaluation of cryptographic primitives that are suitable for devices with constrained resources [3]. The rapid growth of the Internet of Things has led to the emergence of a wide array of exciting, resource-constrained devices that not only sense and interact with their environment but also communicate with other devices within a network. These devices, as part of sensor networks, are vulnerable to various attack vectors, including eavesdropping and data modification during communication. As such, ensuring the integrity and confidentiality of transmitted data is crucial. Traditional cryptographic primitives, however, were not designed with the specific limitations of IoT devices in mind. The capabilities of IoT devices vary significantly, and only those with highly limited resources can be considered truly "lightweight" [2]. For example, Passive RFID Tags, which have minimal computing power, lack independent communication systems, and rely on energy harvesting mechanisms (such as electromagnetic fields) rather than batteries, fall into this category. Cryptographic primitives designed for these devices must, therefore, be specifically developed from the ground up, taking into account their stringent resource constraints.

In contrast, there are more capable devices such as video cameras, drones, and robots, which have significantly different resource requirements. For example, video cameras and drones transmit large volumes of data, which raises important concerns regarding connection jitter and bandwidth, particularly in applications such as video streaming. The development and evaluation of lightweight cryptographic algorithms for secure embedded systems in IoT devices is therefore critical, as conventional cryptographic algorithms may not offer the necessary performance and security for resource-constrained devices. The absence of suitable algorithms often results in insecure system designs, which presents a significant security risk for the expanding network of embedded systems within the IoT ecosystem.

### B. Criteria for Lightweight Algorithms

Embedded systems and IoT devices introduce unique security challenges, and a significant body of research has focused on detecting intrusions within such systems. IoT devices are deployed across a wide range of applications, from consumer monitoring to critical systems like industrial control and highway safety, which has heightened concerns about security and emphasized the need for lightweight security mechanisms. Secure Embedded Systems (SESSs) or IoT devices are typically memory and energy constrained, necessitating lightweight security mechanisms to ensure confidentiality and authenticity.

Traditional security systems often rely on larger data sizes that are unsuitable for embedded systems, highlighting the need for new lightweight cryptographic primitives and their evaluation [3].

Lightweight cryptographic algorithms are specifically designed to protect smaller data sizes in environments with constrained resources, such as Wireless Sensor Networks (WSNs). These algorithms are also platform-independent, making them versatile across various applications. This discussion explores emerging lightweight block and stream ciphers designed to protect data payloads. It begins by examining the criteria for lightweight algorithms, including their internal design parameters, such as block size and data size, which are integral to the overall design of algorithms like AES. The review also addresses the current state of lightweight algorithms, highlighting potential challenges associated with their implementation in simulations and commercial applications.

## VI. EXISTING LIGHTWEIGHT CRYPTOGRAPHIC ALGORITHMS

This section examines lightweight cryptographic algorithms designed for securing embedded systems in IoT devices. A comprehensive review of recently proposed and existing cryptographic algorithms for low-power devices is provided, focusing on their characteristics and performance to better understand the current landscape of lightweight cryptography [2].

A lightweight cryptosystem is proposed for securing various types of data on IoT and pervasive computing devices, which are increasingly popular and widely adopted due to their limited resources. Given the growing importance of protecting the integrity of sensed data or digital information, this paper introduces a lightweight cryptosystem tailored to meet these needs. The Intel Running Average Power Limit (RAPL) interface is utilized for its high-accuracy energy consumption measurement capabilities. A hardware implementation of the system, using Verilog on an FPGA, is provided to demonstrate the performance of the proposed lightweight cryptosystem.

### A. PRESENT

Lightweight cryptography is designed to meet the security requirements of embedded devices with limited physical resources, such as RFID tags, sensors, and actuators. As a result, numerous lightweight cryptographic algorithms have been proposed in recent years. Upon reviewing existing algorithms, they can be categorized into two main groups based on block size: one group features a block size of 64 bits (e.g., PRESENT 128/80, HIGHT 128/128, MISTY1 64/128), while the other group uses a 32-bit block size (e.g., KATAN 32/80, KTANTAN 32/80, Trivium-80) [4]. One such lightweight encryption algorithm with a 64-bit block size is PRESENT.

Cryptographic approaches are essential for enhancing data security against unauthorized access, ensuring confidentiality, integrity, and authenticity of information. Significant efforts have been made to develop cryptographic algorithms that strike a balance between compactness, efficiency, flexibility,

and security, depending on the application. Compact hardware implementations occupy minimal space on the target device, reducing their footprint. In contrast, efficient hardware implementations focus on high throughput per unit area and optimized power consumption, making them suitable for specific applications [2].

### B. SPECK

The Research Center for Smart Artificial Intelligence Systems is an electronics research center founded in 2021, located at Myongji University, South Korea. The Laboratory focuses on developing embedded electronic systems that can handle various environments and maintenance-free using different artificial intelligence (AI)-based techniques and algorithms. To achieve this, miniaturized System on Chip (SoC) designs that utilize-edge AI based on Artificial Neural Network (ANN) and Deep Learning (DL) architectures require implementing efficient algorithms for security purposes in lightweight embedded IoT devices and sensors working in untrusted environments.

The essential properties of a broad lightweight cryptographic algorithm are defined, assuming the relevant goal of protecting embedded systems in IoT devices. The concept and assessment to illustrate the practical use of the SPECK cryptography algorithm are explained. It is a proven lightweight algorithm documented in the International Standards Organization (ISO). Also, its utility is proved by a high-level language application on a field-programmable gate array (FPGA) compared to existing lightweight cryptographic algorithms [3].

### C. SIMON

The SIMON cryptographic algorithm, introduced by the National Security Agency (NSA) in 2013, is widely recognized as a strong candidate for lightweight encryption, particularly in environments with limited computational resources. Its design is well-suited for compact, low-power devices such as wireless sensor nodes, RFID tags, smart meters, smart cards, and other Internet of Things (IoT) applications. SIMON's architecture emphasizes simplicity and efficiency, avoiding complex components like multipliers and look-up tables. The algorithm is available in multiple configurations, supporting block sizes of 48, 64, 96, 128, 192, and 256 bits [3]. In total, there are 22 variants of SIMON, each using key sizes of either  $2n$  or  $n/2$  bits, where  $n$  represents the block size. Notably, all versions share the same round function, contributing to the algorithm's uniformity and ease of implementation. This flexibility makes SIMON a scalable and versatile solution, capable of adapting to a wide range of secure embedded system applications.

Each round function in the SIMON cipher operates using two transformation states, typically denoted as  $x$  and  $y$ . At the end of every round, the  $y$  transformation feeds back into the  $x$  transformation, creating a tightly coupled iterative process. The round function involves a series of lightweight operations, including bitwise AND, XOR, and word rotation. Specifically, the  $x$  transformation applies a right rotation followed by an XOR operation, while the result of the AND

operation is rotated and added to complete the function. These operations are intentionally simple, making SIMON highly efficient for hardware implementations. The architecture of the SIMON cipher family has been thoroughly analyzed, particularly through the hardware exploration of all 11 versions in the SALT\_SIMON benchmark suite [5].

## VII. CHALLENGES IN LIGHTWEIGHT CRYPTOGRAPHY

The development of lightweight cryptographic algorithms introduces new challenges, many of which have not been encountered before. One of the most significant issues in designing such algorithms is the trade-off between security and performance. While most cryptographic algorithms are effective against known attacks, performance concerns often persist [3]. Development efforts are frequently directed towards eliminating bottlenecks such as memory constraints, high power consumption, and algorithmic complexity, which can, ironically, compromise security. As security is often inversely related to performance, these trade-offs must be carefully considered. Algorithm performance should not be evaluated in isolation but rather in conjunction with its cryptographic strength.

Security should always be the primary concern. Miniaturization of devices introduces the risk of new types of attacks, making it essential to anticipate and mitigate these threats during the design process. Striking the right balance is challenging, as it is often easier to attack an algorithm than to defend against it [2]. Many existing attacks are based on theoretical vulnerabilities, which are then exploited in practical implementations. To defend against such threats, developers must consider both the attacker's capabilities and their potential intentions. This may lead to more complex and less power-efficient implementations than necessary. Given the resource-constrained nature of smart embedded devices, lightweight cryptographic algorithms must also be adaptable to various implementation technologies. Should they be implemented in hardware, software, or both? Should different algorithms be designed for different environments? How will decisions regarding implementation technology affect future modifications? A crucial consideration for lightweight algorithms is energy efficiency, especially since they are intended for battery-powered devices where low energy consumption is paramount.

### A. Security vs. Performance Trade-offs

The development of new lightweight cryptographic algorithms must consider both security and performance at every stage of the design process. Typically, designing such algorithms involves balancing the trade-off between these two factors—a balance that depends largely on the required level of security and the type of platform on which the algorithm will be deployed [3]. Enhancing an algorithm's security often comes at the expense of performance. For example, increasing the number of rounds in a block cipher or using a larger key size in a stream cipher can make the encryption more resilient to attacks, but it also raises the cost

of implementation. Conversely, minimizing resource usage by reducing the size or complexity of an algorithm can weaken its security [2]. This inherent tension highlights the need for a careful and systematic evaluation of how performance and security interact. IoT devices, in particular, present a unique challenge: their resource constraints and specialized security needs make them an extreme case of lightweight systems. These limitations significantly shape the design strategies for lightweight cryptographic primitives, often more so than on general-purpose platforms.

### B. Energy Efficiency

Designing devices for IoT requires a strong emphasis on energy efficiency, as these devices must effectively manage limited power resources. To evaluate the energy performance of cryptographic operations, various metrics were employed measuring the energy consumed during encryption and decryption processes in microjoules ( $\mu\text{J}$ ), as well as assessing energy per bit processed and overall power consumption in milliwatts (mW). Accurate energy profiling was achieved using precision tools such as JouleScope and EnergyTrace™, which enabled consistent and repeatable measurements across multiple test iterations. These instruments allowed for a detailed analysis of each cryptographic method under identical operating and environmental conditions, ensuring reliable and comparable results.

## VIII. RESEARCH METHODOLOGY

The methodology for achieving the objectives of this study is structured into four key sequential tasks. The first task focuses on the development of cryptographic algorithms, where three lightweight algorithms based on the Feistel structure are designed and subsequently evaluated. The second task involves the design and implementation of an FPGA-based system-on-chip (SoC), which is utilized to execute cryptographic operations incorporating the developed lightweight algorithms. The third task centers on hardware evaluation, specifically assessing the implementation of access control and data confidentiality systems in the context of IoT security applications. The final task involves modeling and measuring hardware performance metrics, including resource utilization, operating frequency, maximum throughput, and power consumption. These tasks are elaborated upon in the following subsections.

The design of lightweight cryptographic algorithms is grounded in the Feistel network structure, with careful selection of the underlying logic functions to optimize key properties such as security, area efficiency, and processing speed. In the subsequent phase, these lightweight algorithms undergo comprehensive evaluation, which encompasses algorithmic, hardware, and post-implementation assessments. Security evaluation involves various analytical techniques, including differential analysis, linear analysis, and key sensitivity analysis, to assess the robustness of the algorithms against potential attacks. Hardware efficiency is evaluated using logic synthesis tools, with a focus on optimizing for different technologies such as ASIC and FPGA. The algorithms are implemented in

Verilog HDL and examined in terms of resource utilization, maximum operating frequency, throughput per slice flip-flop, and energy consumption per bit.

The system design is developed using Xilinx FPGAs, where the processor is integrated with various peripherals. Lightweight cryptographic operations, including encryption, authentication, and key generation, are executed by dedicated peripherals with separate interfaces from the processor. This design ensures the security of the data bus between the processor and peripherals, as well as the data transfers between external memory and peripherals. The system is implemented on the Xilinx Virtex 5 ML501 FPGA development board, and its functionality is verified through simulation and logic analyzer testing to ensure proper operation and performance.

In this task, the performance of the hardware is evaluated by implementing the lightweight algorithms across various hardware platforms. The assessment focuses on key performance metrics, including resource utilization, power consumption, throughput, operating frequency, and energy per bit. These indicators provide a comprehensive analysis of the system's efficiency and effectiveness in handling cryptographic operations.

#### A. Algorithm Design and Evaluation

The increasing reliance on IoT devices has heightened the need for robust security and privacy measures. Given the risk of sensitive data falling into the wrong hands, there is a pressing demand for lightweight cryptographic enhancements tailored to IoT environments. This section outlines the research methods employed to address this challenge. First, the criteria for developing lightweight cryptographic algorithms will be discussed. Subsequently, the devices utilized in the study will be tested, and the simple chaos-based stream cipher algorithm, developed as part of this research, will be evaluated against established standards. The findings will be analyzed to ensure that they meet the necessary security and efficiency criteria.

In designing cryptographic algorithms for embedded systems such as IoT devices, careful attention must be paid to both algorithmic complexity and resource utilization. The computational cost of a cryptographic algorithm is typically measured in terms of basic operations such as multiplications, XORs, and shifts. Performance metrics, such as execution time and area requirements, are also crucial to assess the feasibility of implementing the algorithm in resource-constrained environments.

#### B. Performance Metrics

Performance metrics are essential in assessing the effectiveness of cryptographic algorithms, offering both quantitative and qualitative measures to characterize their efficiency. This section focuses on the specific performance metrics used to evaluate the proposed cryptographic algorithms.

**Computational Speed:** Computational speed refers to the time required by a cryptographic algorithm to encrypt and decrypt data. It is commonly expressed in terms of throughput

or data rate, where higher throughput indicates greater efficiency of the algorithm. To evaluate computational speed, data encryption and decryption times are measured using TestU01, with input data sizes varying from 512 bits to 2048 bits. The throughput is calculated using the following formula: [2]

$$\text{Throughput (bits/second)} = N / T$$

Where:

- N is the number of plaintext bits.
- T is the time taken in seconds for encryption

Memory utilization refers to the amount of RAM and ROM required for the cryptographic algorithm to operate on a platform. In the context of embedded systems and IoT devices with limited resources, an ideal cryptographic algorithm should be designed to minimize memory usage while still providing effective security. The goal is to achieve efficient performance with the smallest memory footprint, particularly for resource-constrained devices. To measure memory utilization, the cyclical-based random MAC (scr-MAC) and the memory utilization of the proposed algorithms are evaluated using the FELICS framework. This framework is designed to assess and compare the memory usage of lightweight cryptographic systems. While more complex algorithms can be developed, they may inadvertently increase memory usage, which is undesirable for embedded systems with limited memory capacity. Thus, the focus is on utilizing smaller memory sizes while maintaining strong security features.

The memory utilization of the proposed algorithms is as follows:

- 3-round implementation of a chaotic stream cipher (scr-CMAC): 61.7 bytes.
- MAC-based cryptosystems: 159.84 bytes.

In comparison to other lightweight designs, which typically require fewer than 500 gates, these algorithms demonstrate efficient memory utilization while still offering effective protection against various types of attacks. This approach ensures that the proposed algorithms remain practical for use in constrained environments.

### IX. CASE STUDIES AND APPLICATIONS

This section presents a series of case studies that illustrate methodologies for developing and evaluating lightweight cryptographic algorithms tailored for secure IoT environments. These examples demonstrate practical implementations and research-oriented developments in domains such as embedded system security and smart grid infrastructures.

A secure and lightweight encryption/decryption cryptosystem is proposed, specifically designed for pervasive computing and IoT devices. The design criteria for the cryptographic primitives include the absence of trust dependencies, platform compatibility, low computational complexity, and high performance [2]. The system incorporates homomorphic encryption techniques at two security levels: integer-based and floating-point-based. The integer-level encryption is constructed using lattice structures and Lotkin functions. Comprehensive theoretical security analyses are conducted for both cryptographic schemes.

In the context of smart grids and smart metering infrastructures, which rely on bidirectional communication in daily energy consumption systems, ensuring privacy and data security is critical. The widespread deployment of smart meters and connected appliances in home automation increases the risk of unauthorized data access and misuse.

To address these challenges, an architectural framework is proposed that safeguards data and user privacy while supporting smart home automation functionality. The framework identifies areas where commercially available products can be leveraged, and highlights configurations that enhance security capabilities on the consumer end of smart metering systems. A literature review identifies prevalent attack vectors and outlines key security considerations that developers and manufacturers should integrate at the design stage. Additionally, a system is proposed that mitigates identity-disclosure risks by storing sensitive information outside of the local network environment.

#### A. IoT Security

As the number and diversity of IoT nodes and their associated networks continue to expand, security solutions must be carefully reconsidered, adapted, and redesigned to address the specific constraints and unique characteristics of these environments [2]. Traditional security mechanisms originally developed for computer-centric infrastructures have often proven inadequate when applied to decentralized, peer-to-peer systems and networks comprising devices with limited processing capabilities. Moreover, legacy security frameworks, which typically rely on assumptions tied to particular hardware or operating system configurations, are increasingly vulnerable to evolving threats, including malware, worms, denial-of-service (DoS) attacks, and internet-based automated bots. These shifts in architecture and deployment context have introduced new attack surfaces and vulnerabilities, underscoring the urgent need for lightweight, context-aware, and resilient security paradigms tailored to modern IoT ecosystems.

The emergence of what is increasingly regarded as the next generation of the World Wide Web often referred to as the "New Web" is characterized by a dramatic expansion in the diversity and volume of data sources and data intelligence embedded within networks of interconnected devices, sensors, and computing systems. Unlike traditional architectures, which rely primarily on internal, electronically governed computer nodes as data sources, this paradigm shift introduces more heterogeneous and dynamic data flows. As a result, the nature of marketable data has evolved, significantly increasing the complexity and scope of required security countermeasures [1]. In response to these challenges, particularly within constrained, resource-limited, and operationally diverse environments, this research focuses on the development and application of lightweight cryptographic solutions. These algorithms are specifically tailored to peer-to-peer frameworks, aiming to minimize device overhead while maintaining confidentiality and mitigating network vulnerabilities.

#### B. Smart Grids

Smart grids, which integrate digital technologies for enhanced energy management, face significant security and privacy challenges due to the interconnectedness of devices like smart meters and sensors. Lightweight cryptographic algorithms are essential for securing these devices, as they operate with limited resources such as memory, processing power, and energy. The algorithms are validated using ASIC and FPGA hardware architecture simulations on a testbed with different specifications found in the actual implementation of SG's devices. The results indicate that the proposed methods and hardware implementations enable successful surveillance of SG's sensors and control the operating conditions, regardless of the implementation technology [1].

### X. FUTURE DIRECTIONS AND EMERGING TRENDS

The vulnerability of current public-key cryptographic algorithms to quantum attacks has become a focal point in contemporary cryptographic research. A recent report by the National Security Agency (NSA) warns that quantum computers capable of breaking widely used public-key systems may become available within the next decade. In anticipation of this threat, the cryptographic community has turned its attention to the development of quantum-resistant, also referred to as post-quantum or quantum-safe, public-key algorithms. These include both lattice-based and non-lattice-based approaches, several of which are undergoing standardization through the National Institute of Standards and Technology (NIST). Notably, many proposals include both standard and lightweight variants to address different implementation environments. Among the most prominent and widely recommended lightweight cryptographic schemes are those developed by the CLoCK and NTRU initiatives, which continue to shape the landscape of post-quantum cryptographic design.

The CLoCK framework represents a suite of lightweight cryptographic primitives built upon the PRESENT block cipher, employing a feedback-with-filter methodology. It introduces several key and stream generator designs, including LST, GFT, RGS, and PGS, to address various lightweight encryption scenarios. In parallel, NTRU comprises a family of polynomial-based public-key cryptographic schemes, developed since 1996, which rely on hard problems defined over polynomial rings. A recent advancement in this domain is the NTRU-HRSS-based data protection mechanism, which incorporates HELLO padding and cryptographic hash functions such as SHA-256 or M320T. This solution presents a promising alternative for secure communication in embedded and Internet of Things (IoT) environments. Ongoing efforts are focused on improving community access to CLoCK and NTRU primitives through integration into modern, user-friendly smart card platforms, thereby enabling practical deployment of post-quantum lightweight cryptography in real-world applications.

### A. Post-Quantum Lightweight Cryptography

As a natural progression in the study of lightweight cryptographic systems, attention is increasingly turning toward the integration of post-quantum lightweight cryptography. Recent advances in quantum computing have intensified concerns about the long-term viability of current cryptographic standards, particularly those based on public-key mechanisms. Much like the historical shift from classical to public-key cryptography, which was prompted by evolving computational capabilities, the anticipated rise of practical quantum computers is expected to drive a transition toward post-quantum cryptographic solutions. Notably, quantum algorithms such as Shor's algorithm threaten the security of widely adopted asymmetric cryptosystems, including RSA and elliptic curve cryptography (ECC), under conventional security assumptions [6]. In response, there is an increasing imperative to assess the feasibility and performance of quantum-resistant algorithms, especially within resource-constrained environments where lightweight cryptography is essential.

In the continued exploration of lightweight cryptographic algorithms, increasing emphasis is being placed on post-quantum lightweight cryptography, particularly in light of the quantum resistance properties of the evaluated schemes. Potential design directions include lattice-based approaches and probabilistic or degree-based extensions of classical lightweight algorithms. Special attention is given to the energy efficiency and overall performance of these quantum-resistant cryptographic primitives when deployed in resource-constrained environments. In response to the growing need for quantum-secure solutions, the National Institute of Standards and Technology (NIST) has initiated a multi-phase standardization process to evaluate and select post-quantum cryptographic algorithms, aligned with the ISO/IEC 19790 framework. As the field awaits the outcome of the fourth round of this process, there is significant interest in assessing the current status of candidate algorithms suitable for embedded systems. Key performance metrics—particularly those involving the rate of public-key and private-key operations per second—are being scrutinized to determine their suitability for real-world deployment [2]. Parallel to this effort, the standardization of lightweight cryptographic primitives is also progressing through ISO/IEC JTC 1/SC 27 WG 2, which focuses on defining cryptographic solutions optimized for environments with limited computational resources. As part of this initiative, a Call for Nominations was issued in March 2017 for a New Work Item Proposal based on NIST SP 800-185, outlining a profile of lightweight cryptographic primitives tailored for Internet of Things (IoT) applications. Recent developments concerning these standardization efforts continue to shape the landscape of lightweight and post-quantum cryptography.

## XI. CONCLUSION AND RECOMMENDATIONS

This study highlights the growing importance of lightweight cryptographic algorithms in securing embedded systems within IoT infrastructures. The research involved the development of new block ciphers designed for low-power

consumption, minimal memory usage, and efficient processing—key considerations for resource-limited devices. Through rigorous hardware implementation and testing using FPGA platforms, the proposed algorithms demonstrated strong performance metrics, including high throughput and low energy consumption. Comparative evaluations show that algorithms like the HIGHT block cipher and SHA-3 hash function can be effectively deployed on 8-bit microcontrollers with minimal overhead. Furthermore, the findings underscore the need to continue refining cryptographic solutions to prepare for future threats, such as quantum computing. Ultimately, this work contributes valuable insights for researchers and developers aiming to implement secure, efficient, and scalable cryptographic systems in the ever-expanding landscape of IoT and embedded technologies.

### A. Summary of Findings

Cryptography is a crucial field that allows the creation of secure systems for private communication. Data Encryption Standard (DES) is the most generally used symmetric key algorithm by companies. As the need for data security rises, companies throughout the globe are implementing robust protection strategies based on advanced Encryption Standards (AES). With the rise of the Internet of Things (IoT) ideas, security becomes increasingly crucial owing to the inclusion of numerous resource-constrained devices in smart cities. Cryptographic algorithms acting on the data type in the same group can secure the data from being compromised. In this regard, there is an increasing demand for lightweight cryptographic algorithms for safe embedded systems in IoT devices. This research study focuses on the creation and assessment of several lightweight cryptographic algorithms for different types of data in an IoT setting. The modern cryptographic methods offer protection in terms of various cryptanalysis, and performance. The resource usage of all the created algorithms is evaluated and confirmed in FPGA hardware as necessary in a smart city environment [3].

The main impact of this study may be stated in the following manner: Development of a new lightweight block cipher method decreased the use of energy is a fundamental component of lightweight systems [2]. Basic mathematical ideas guiding the new cryptosystem such as composite residue function and mapping functions are optimized for least amount of mathematical operations. Systematic testing of the cryptosystem in terms of unconditional security is recommended, in particular concentrating on broad threats such brute force search on a device implementing the lightweight public key cryptosystem.

### B. Practical Implications

The lightweight cryptographic algorithms developed and evaluated in this study are well-suited for researchers and practitioners working on secure embedded systems in IoT devices. This research provides valuable insights into the feasibility and performance of the SHA-3 hash function and the HIGHT block cipher on an 8-bit 8051 microcontroller,

carefully considering constraints like limited memory and processing cycles. Notably, the HIGHT cipher can be implemented using less than 1 KB of code memory, and the SHA-3 hash function—when using the resource-efficient cSHAKE variant—also fits within the same memory footprint. These results highlight the practicality of using such algorithms in resource-constrained environments. Researchers and system designers are encouraged to adopt and adapt these lightweight cryptographic solutions to meet the specific requirements of their embedded systems.

The VHDL implementations of the SHA-3 hash function and HIGHT block cipher algorithm developed in this work are well-suited for researchers and system developers working on secure embedded systems—particularly in IoT applications deployed in tropical rainforest environments. Evaluation results indicate that the HIGHT block cipher is more resource-efficient than SHA-3, requiring fewer hardware resources, making it especially advantageous for hardware-constrained devices. These implementations target a 1K LUT, 4-input FPGA architecture and offer a strong foundation for developing lightweight security solutions. Furthermore, the designs can be adapted for system-on-chip (SoC) architectures, enabling integration onto a single die. This integration not only enhances performance but also helps reduce the cost of implementing these cryptographic algorithms in real-world embedded systems.

#### REFERENCES

- [1] M. Abu-Tair, S. Djahel, P. Perry, B. Scotney et al., "Towards Secure and Privacy-Preserving IoT Enabled Smart Home: Architecture and Experimental Study," 2020.
- [2] M. Abutaha, B. Atawneh, L. Hammouri, and G. Kaddoum, "Secure lightweight cryptosystem for IoT and pervasive computing," 2022.
- [3] M. Rana, Q. Mamun, and R. Islam, "Current Lightweight Cryptography Protocols in Smart City IoT Networks: A Survey," 2020.
- [4] N. A. Gunathilake, A. Al-Dubai, W. J. Buchanan, and O. Lo, "Electromagnetic Analysis of an Ultra-Lightweight Cipher: PRESENT," 2021.
- [5] A. Shahverdi, "Lightweight Cryptography Meets Threshold Implementation: A Case Study for SIMON," 2015.
- [6] G. Banegas, K. Zandberg, A. Herrmann, E. Baccelli et al., "Quantum-Resistant Security for Software Updates on Low-power Networked Embedded Devices," 2021.

# Advanced Machine Learning and Swarm Intelligence for Enhanced Cyanobacterial Bloom Prediction: A Comparative Study

Bounekhla Oumaima

Dept. of Computer Science,

Badji Mokhtar University Annaba, Algeria  
bounekhlaoumaima@gmail.com

Hemici Meriem

Dept. of Computer Science,

Badji Mokhtar University Annaba, Algeria  
hemicimeriem@gmail.com

N. Dendani

Dept. of Computer Science,

Badji Mokhtar University Annaba, Algeria  
ndendani@yahoo.fr

Amel Saoudi

Dept. of Biochemistry,

Badji Mokhtar University Annaba, Algeria  
amelsaoudi@yahoo.fr

Nabiha Azizi

Dept. of Computer Science,

Badji Mokhtar University Annaba, Algeria  
nabiha111@yahoo.fr

**Abstract**—Cyanobacterial harmful algal blooms (CyanoHABs) pose significant threats to freshwater ecosystems, water quality, and public health, particularly in regions affected by climate change and nutrient enrichment. Accurate forecasting of these blooms is essential for proactive water resource management and mitigation efforts. This study proposes a robust and scalable machine learning framework for CyanoHAB prediction, leveraging advanced ensemble models optimized via Particle Swarm Optimization (PSO) for hyperparameter tuning.

We evaluate several state-of-the-art algorithms, including Random Forest (RF), XGBoost, LightGBM, CatBoost, and HistGradientBoosting, while also exploring hybrid ensemble architectures to enhance predictive accuracy. The methodology is applied to two contrasting datasets: (1) a dataset from the Mexa Dam in Algeria, augmented using a Conditional Generative Adversarial Network (CGAN) to address data scarcity, and (2) a large-scale dataset from multiple South Korean rivers (Nakdong, Han, and Geum), which are prone to frequent CyanoHAB occurrences.

Comprehensive data preprocessing was conducted, including K-Nearest Neighbors (KNN) imputation and domain-informed feature engineering to capture key environmental drivers. Experimental results show that the PSO-optimized Random Forest achieves outstanding performance, with an  $R^2$  score of 0.9993, a mean absolute error (MAE) of 0.1764, and a root mean squared error (RMSE) of 0.2895. Furthermore, a stacked ensemble combining LightGBM and HistGradientBoosting with a linear meta-learner achieves an exceptional  $R^2$  score of 0.9999, confirming the effectiveness of hybrid modeling. The impact of CGAN-based data augmentation is also validated, showing improved generalization on sparse datasets.

This work provides three main contributions: (1) a high-accuracy AI framework for CyanoHAB forecasting, (2) a comparative analysis of ensemble and hybrid models, and (3) actionable insights for ecological monitoring across diverse hydrological systems. These findings demonstrate the potential of AI-driven methods to support sustainable water management and enhance public health resilience.

**Index Terms**—Cyanobacterial Harmful Blooms (CyanoHABs), Machine Learning (ML), Particle Swarm Optimization (PSO), Random Forest (RF), XGBoost, LightGBM, CatBoost, HistGradientBoosting, Ensemble Learning, Water Quality Prediction, Harmful Algal Blooms (HABs), Nakdong River, Han River, Geum River, Mexa Dam, Data Augmentation, Conditional Generative

Adversarial Network (CGAN), Swarm Intelligence, Hyperparameter Optimization, Environmental Modeling, Freshwater Ecosystems, Predictive Modeling, Time Series Analysis, Water Resource Management.

## I. INTRODUCTION

Cyanobacterial harmful blooms (CyanoHABs) represent one of the most pressing environmental challenges of the 21st century, threatening freshwater ecosystems, water quality, and public health on a global scale. These blooms, characterized by the rapid proliferation of cyanobacteria, are often fueled by eutrophication—excessive nutrient loading from agricultural runoff, industrial discharge, and urban wastewater—coupled with the effects of climate change, such as rising temperatures and altered precipitation patterns [1], [2]. Cyanobacteria produce a range of toxins, including microcystins, anatoxins, and saxitoxins, which pose serious risks to human health, causing liver damage, neurological disorders, and skin irritation [3], [4]. Additionally, CyanoHABs have significant economic impacts, affecting water treatment costs, fisheries, tourism, and recreational activities [5], [6].

In Algeria, CyanoHABs have gained increasing attention due to recurrent blooms observed in key water bodies, such as the Mexa Dam. This reservoir, a critical water resource for agriculture and domestic use, has experienced frequent cyanobacterial outbreaks, exacerbated by nutrient pollution and climatic stressors [7]. Similarly, in South Korea, several major rivers, including the Nakdong, Han, and Geum Rivers, have faced persistent CyanoHABs, particularly in their downstream regions. The construction of multifunctional weirs along these rivers, aimed at improving water quality and flood control, has inadvertently increased water residence times, creating favorable conditions for cyanobacterial growth [8], [9]. These case studies highlight the urgent need for effective prediction and management strategies to mitigate the impacts

of CyanoHABs in diverse geographical and environmental contexts.

Traditional approaches to predicting CyanoHABs have relied on mechanistic models, such as the Environmental Fluid Dynamics Code (EFDC) and Delft3D, which simulate water quality parameters and algal dynamics based on physical, chemical, and biological processes [10]. While these models provide valuable insights into the mechanisms driving CyanoHABs, they often require extensive computational resources, detailed input data, and are limited by uncertainties in parameter estimation and environmental variability [10]. Moreover, mechanistic models are typically site-specific and may not generalize well to other regions or water bodies, limiting their applicability for large-scale water resource management.

In recent years, machine learning (ML) approaches have emerged as powerful alternatives for predicting CyanoHABs, offering the ability to handle complex, high-dimensional datasets and capture nonlinear relationships between environmental drivers and bloom dynamics [11], [12]. Techniques such as Random Forest (RF), XGBoost, LightGBM, and CatBoost have demonstrated remarkable success in water quality prediction and algal bloom forecasting, often outperforming traditional models in terms of accuracy and computational efficiency [13], [14].

Furthermore, ensemble learning approaches, which combine multiple models, have shown enhanced performance by leveraging the strengths of individual algorithms [15]. Recent advancements in swarm intelligence, such as Particle Swarm Optimization (PSO), have further improved model performance by optimizing hyperparameters and reducing prediction errors [16]. In this study, we extend these approaches by exploring innovative **model combinations**, such as **XGBoost + Random Forest** and **LightGBM + HistGradientBoosting**, to achieve superior prediction accuracy.

In this study, we present a comprehensive framework for predicting CyanoHABs by integrating machine learning models and swarm intelligence techniques. We utilize two distinct datasets: (1) A dataset from the Mexa Dam in Algeria, augmented using a Conditional Generative Adversarial Network (CGAN) to enhance data quality and quantity. (2) A dataset from multiple South Korean rivers, including the Nakdong, Han, and Geum Rivers, spanning from 2005 to February 2025 and covering diverse water bodies such as rivers and lakes. The South Korean dataset includes detailed water quality parameters, such as temperature, pH, dissolved oxygen (DO), turbidity, and chlorophyll-a, as well as cyanobacterial cell counts for genera including *Microcystis*, *Anabaena*, *Oscillatoria*, and *Aphanizomenon*.

Our approach evaluates and compares the performance of various ML models (RF, XGBoost, LightGBM, CatBoost, HistGradientBoosting), all optimized using PSO. Additionally, we explore ensemble techniques, combining models to achieve superior prediction accuracy. For instance, the combination of **LightGBM and HistGradientBoosting**, followed by stacking with linear regression, achieves an exceptional **R<sup>2</sup> score of**

**0.9999**, demonstrating the power of hybrid ensemble approaches.

The primary objectives of this study are:

- 1) To develop a robust predictive framework for CyanoHABs using advanced ML, swarm intelligence, and hybrid ensemble techniques.
- 2) To compare the performance of different models and their combinations in predicting cyanobacterial cell concentrations across diverse environments.
- 3) To provide actionable insights for water resource management and bloom mitigation strategies in both Algeria and South Korea.

By integrating data from two geographically and environmentally distinct regions, this study highlights the adaptability and scalability of AI-driven approaches for CyanoHAB prediction. The combination of data augmentation (CGAN), optimization techniques (PSO), and ensemble learning further enhances the robustness and accuracy of our predictive models. This work not only contributes to the scientific understanding of CyanoHAB dynamics but also provides practical tools for water management authorities to anticipate and mitigate bloom events effectively. Ultimately, our findings aim to support sustainable water resource management and public health protection in regions vulnerable to CyanoHABs.

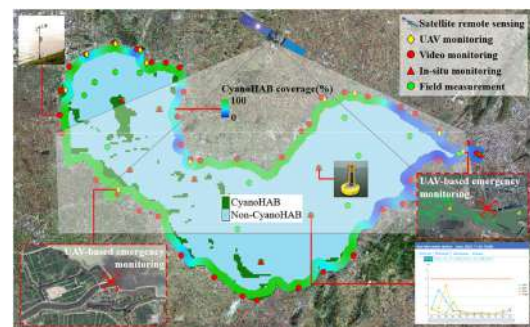


Fig. 1. Monitoring approaches for CyanoHABs in the Nakdong River, South Korea. Techniques include satellite remote sensing, UAV-based monitoring, in-situ measurements, and video surveillance, providing comprehensive surveillance for effective bloom management.

As shown in Figure 1, the monitoring approaches for CyanoHABs in the Nakdong River include a combination of advanced technologies such as satellite remote sensing, UAV-based monitoring, in-situ measurements, and video surveillance. These techniques provide a comprehensive surveillance system that enhances the ability to predict and manage cyanobacterial blooms effectively.

## II. RELATED WORK

### A. Cyanobacterial Blooms in Eastern Algeria

Since the 1990s, cyanobacterial blooms have been a significant concern in Eastern Algeria, particularly in the Mexa Reservoir. [7] identified seven cyanobacterial genera in this reservoir, five of which are potentially toxic, and established strong correlations between cyanobacterial density,

chlorophyll-a concentrations, and microcystin levels. These findings highlight the urgent need for effective monitoring and prediction systems to mitigate the impacts of CyanoHABs on water quality and public health in the region. Despite these efforts, traditional monitoring methods remain limited in their ability to provide real-time, accurate predictions, underscoring the necessity for advanced computational approaches.

#### B. Traditional Approaches for CyanoHAB Prediction

Traditional methods for predicting CyanoHABs have relied on **mechanistic models** such as the Environmental Fluid Dynamics Code (EFDC) and Delft3D. These models simulate water quality parameters and algal dynamics based on physical, chemical, and biological processes. While they provide valuable insights, they are often computationally intensive, require extensive input data, and are limited by uncertainties in parameter estimation and environmental variability [?]. Additionally, their site-specific nature limits their applicability to other regions or water bodies.

#### C. Machine Learning Approaches for CyanoHAB Prediction

In recent years, **machine learning (ML)** has emerged as a powerful alternative for CyanoHAB prediction due to its ability to handle complex, high-dimensional datasets and capture nonlinear relationships. Early studies employed **artificial neural networks (ANNs)** to predict phytoplankton and *Microcystis* biomass dynamics, achieving accuracies up to 0.97 [?]. More recently, [13] compared multiple ML models, including Decision Trees, K-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forest (RF), with RF achieving an accuracy of 95.81% for algal bloom prediction.

**Ensemble learning** approaches have further enhanced prediction accuracy by combining multiple models. For example, [15] demonstrated the effectiveness of ensemble models in predicting cyanobacterial concentrations, achieving  $R^2$  values exceeding 0.95. However, these studies often lack optimization techniques to fine-tune model performance, leaving room for improvement.

#### D. Advanced Techniques: Deep Learning and Hybrid Models

Advanced deep learning architectures have significantly improved CyanoHAB prediction accuracy. [19] developed an integrated **CNN-LSTM model** to predict CyanoHABs in Taihu Lake, achieving an  $R^2$  of 0.91. In 2023, [20] integrated **CNN and Transformer algorithms**, with the CNN-Transformer model exhibiting superior performance. These approaches demonstrate the potential of hybrid architectures for capturing complex spatiotemporal patterns in bloom dynamics.

In 2024, [16] developed a **Bayesian network** to predict cyanobacterial blooms two weeks in advance, achieving an area under the curve (AUC) of 0.83. Similarly, [17] applied ML and DL techniques to forecast *Aureococcus anophagefferens* population density, with all models (RF, SVR, MLP, CNN) achieving  $R^2$  values exceeding 0.75. These studies highlight the growing trend of combining ML with domain-specific knowledge for improved prediction accuracy.

#### E. Gaps and Contributions of Our Study

While previous studies have made significant contributions, several gaps remain:

- **Limited Generalization:** Many models are site-specific and lack adaptability to diverse environmental contexts.
- **Insufficient Optimization:** Few studies explore advanced optimization techniques like **Particle Swarm Optimization (PSO)** to enhance model performance.
- **Data Scarcity:** The lack of high-quality, diverse datasets limits the development of robust prediction frameworks.

#### Our study addresses these gaps by:

- Evaluating a wide range of ML models, including **RF, XGBoost, LightGBM, CatBoost, HistGradientBoosting, TPOT, and XGBRF**, and optimizing them using **PSO**.
- Exploring innovative **ensemble combinations** (e.g., RF + XGBoost, LightGBM + HistGradientBoosting) to achieve state-of-the-art performance.
- Utilizing two distinct datasets—from the **Mexa Dam in Algeria** (augmented using **CGAN**) and **South Korean rivers** (Nakdong, Han, Geum)—to demonstrate the adaptability of our approach.
- Providing a scalable framework for water resource management authorities to anticipate and mitigate CyanoHAB impacts effectively.

By integrating **ensemble learning, swarm intelligence, and hybrid modeling**, our work advances the state of the art in CyanoHAB prediction and offers practical tools for sustainable water management.

### III. MATERIALS AND METHODS

#### A. System Architecture

Here is a visual representation of the complete architecture of the CyanoHAB prediction system, including preprocessing, models, optimization, and evaluation:

#### B. Site Description

1) *Mexa Dam (Algeria):* The Mexa Dam is located in the commune of Bougous, within the wilaya of El Tarf, Algeria, at the Gorge de Mexenne (36°45'14.31"N, 8°23'33.68"E). This reservoir is a critical freshwater resource for the region, supporting both ecological and human needs. However, its hydrological and ecological characteristics make it susceptible to cyanobacterial blooms, which can have significant environmental and health impacts.

To monitor these conditions, data were collected from ten monitoring stations over a 24-month period, from January 2010 to December 2011. Monthly sampling was conducted, resulting in a total of 240 samples. The dataset comprises 17 features, categorized into biotic, abiotic, nutritional, physical, and meteorological factors, as detailed in Table I.

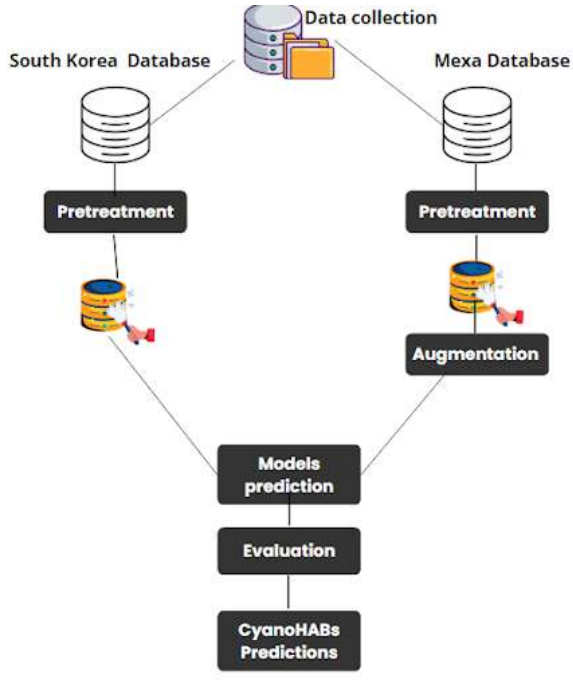


Fig. 2. Complete architecture of the CyanoHAB prediction system.

a) *Application of GAN and CGAN:* To enhance the dataset and address the challenges of limited data availability, Generative Adversarial Networks (GANs) were applied. Initially, a standard GAN was used to generate synthetic data. However, the results were suboptimal, as the generated data appeared too random and did not adequately reflect the statistical properties of the real dataset. This limitation is common in GANs when the model fails to capture the underlying distribution of the data.

To overcome this issue, a Conditional Generative Adversarial Network (CGAN) was implemented. Unlike a standard GAN, a CGAN conditions the data generation process on specific features, allowing for more controlled and realistic synthetic data generation. By incorporating additional contextual information (e.g., environmental conditions, time of year), the CGAN produced synthetic data that closely matched the real dataset in terms of statistical properties and patterns. This approach significantly improved the quality of the generated data, making it suitable for further analysis and modeling.

2) *Rivers in South Korea:* The South Korean rivers dataset encompasses data from three major rivers: the Nakdong, Han, and Geum Rivers. This dataset spans a 20-year period, from 2005 to 2025, and includes over 20,000 samples collected from various rivers, lakes, and other water bodies. The dataset is particularly valuable for long-term ecological and hydrological studies, as it provides insights into the dynamics of cyanobacterial blooms and water quality over time.

The dataset includes features categorized into biotic, abiotic, and meteorological factors, as detailed in Table II.

TABLE I  
FEATURE DESCRIPTION FOR MEXA DAM DATASET

Category	Feature Name	Data Type	Range/Values
Biotic	Cyanobacteria density (Cell/ml)	Numerical	0 - 1.12×10 <sup>7</sup>
	Chlorophyll-a (µg/l)	Numerical	0 - 180
	Microcystins (µg/l)	Numerical	0 - 168.41
Abiotic	Water temperature (°C)	Numerical	1.3 - 31.1
	pH	Numerical	6.26 - 9.98
	Dissolved oxygen (mg/l)	Numerical	2.41 - 15.04
	Conductivity (µs/cm)	Numerical	389 - 648
	Salinity (g/l)	Numerical	0.1 - 0.38
Nutritional	Turbidity (NTU)	Numerical	0.24 - 113.3
	Nitrates (mg/l)	Numerical	0.1 - 34.62
	Nitrites (mg/l)	Numerical	0 - 1.6
	Ammoniums (mg/l)	Numerical	0.001 - 2.2
	Phosphorus (mg/l)	Numerical	0.001 - 1.2
Physical	Depth (m)	Numerical	0 - 18
	Station's location	Categorical	Exposed, Sheltered, No role for wind
Meteorological	Air temperature (°C)	Numerical	1.3 - 24.0

TABLE II  
FEATURE DESCRIPTION FOR SOUTH KOREA RIVERS DATASET

Category	Feature Name	Data Type	Ranges/Values
Biotic	Chlorophyll (a) (µg/l)	Numerical	10.2 - 17.5
	Cyanobacteria's density (Cell/ml)	Numerical	0 - 22
	Microcystis (Cell/ml)	Numerical	0 - 0
	Anabaena (Cell/ml)	Numerical	0 - 0
	Oscillatoria (Cell/ml)	Numerical	0 - 0
	Aphanizomenon (Cell/ml)	Numerical	0 - 22
Abiotic	Water temperature (°C)	Numerical	2.5 - 7.0
	pH	Numerical	7.9 - 9.0
	Dissolved oxygen (mg/l)	Numerical	13.9 - 15.8
	Transparency (m)	Numerical	1.2 - 2.0
	Turbidity (NTU)	Numerical	1.5 - 2.7
Meteorological	Date	Categorical	Various timestamps
	Site	Categorical	'Nakdonggang', others
	Location	Categorical	'Haepyeong', others

### C. Data Preprocessing

1) *Data Integration:* The datasets from the **Mexa Dam (Algeria)** and the **South Korean rivers** were prepared for analysis separately. Each dataset underwent a rigorous cleaning process to handle missing values, inconsistencies, and outliers. This step ensured the reliability and quality of the data before feeding it into machine learning models.

2) *Handling Missing Values:* Missing values were addressed using a two-step approach to ensure data quality and reliability:

- **Removal of Rows with Excessive Missing Values:** Rows with more than 10 missing attributes were identified and removed from the dataset. This step was necessary to eliminate incomplete records that could introduce bias or reduce the effectiveness of the imputation process. After this removal, only 4 rows with 4 missing attributes remained.
- **K-Nearest Neighbors (KNN) Imputation:** The remaining missing values were imputed using the K-Nearest Neighbors (KNN) method. This approach was chosen because it preserves the relationships between variables by estimating missing values based on the values of the nearest neighbors. KNN imputation is particularly effective for datasets with complex interdependencies, as it leverages the local structure of the data to fill in gaps.

The KNN imputation process involved the following steps:

- **Normalization:** All features were normalized to ensure that distances between data points were calculated on a comparable scale.

- **Distance Calculation:** The Euclidean distance was used to identify the  $k$  nearest neighbors (where  $k$  was chosen based on cross-validation).
- **Imputation:** The missing values were replaced with the weighted average of the corresponding values from the nearest neighbors.

This two-step approach ensured that the dataset remained robust and suitable for machine learning models, while minimizing the impact of missing data on the analysis.

3) *Feature Engineering:* To prepare the data for machine learning models, several feature engineering steps were applied:

- **Categorical Encoding:** The categorical variable *Station's location* in the Mexa Dam dataset was encoded using one-hot encoding. This transformation converts categorical data into a numerical format, making it suitable for machine learning algorithms.
- **Temporal Feature Extraction:** The *Date* column in the South Korean rivers dataset was converted to a datetime format to facilitate time-series analysis. Additional temporal features, such as month, season, and day of the year, were extracted to capture seasonal and temporal patterns in cyanobacterial blooms.
- **Feature Scaling:** To ensure that all features contribute equally to the model's performance, standardization was applied. Each numerical feature was transformed to have a mean of 0 and a standard deviation of 1. This step is crucial for models sensitive to the scale of input data, such as support vector machines (SVM) and neural networks. The standardization formula used was:

$$z = \frac{x - \mu}{\sigma}$$

where  $x$  is the original value,  $\mu$  is the mean, and  $\sigma$  is the standard deviation.

- **Outlier Handling:** Outliers were identified using the Interquartile Range (IQR) method. Values outside the range  $[Q1 - 1.5 \times IQR, Q3 + 1.5 \times IQR]$  were considered outliers and replaced with the nearest boundary value to minimize their impact on the model.
- **Feature Selection:** To reduce dimensionality and improve model efficiency, feature importance was evaluated using Random Forest. Features with low importance scores were removed to simplify the model without sacrificing predictive accuracy.

4) *Data Augmentation (Mexa Dam Dataset):* To address the limited size of the Mexa Dam dataset, data augmentation was performed using a **Conditional Generative Adversarial Network (CGAN)**. The CGAN generated synthetic data that closely resembled the real dataset, enhancing the model's ability to generalize to unseen data. This step was particularly important for improving the performance of machine learning models on smaller datasets. Before augmentation, the dataset underwent normalization and other preprocessing steps to ensure consistency and quality.

5) *Dataset Splitting:* The preprocessed datasets were split into training, validation, and test sets using an **80-10-10 split**. This division ensured that the models were trained on a sufficient amount of data while retaining separate datasets for validation and testing to evaluate their performance objectively.

#### D. Models

1) *Ensemble Machine Learning Models:* This study explored the prediction of **CyanoHABs (Cyanobacterial Harmful Algal Blooms)** in freshwater ecosystems using a suite of advanced **ensemble learning models**, including **Random Forest (RF)**, **XGBoost**, **CatBoost**, **LightGBM**, **TPOT**, and **HistGradientBoosting**. Each model was optimized using **Particle Swarm Optimization (PSO)** to fine-tune hyperparameters, ensuring maximum predictive accuracy and robustness.

- **Random Forest (RF):** Used as a baseline model, RF leverages an ensemble of decision trees to enhance predictive accuracy and reduce overfitting. It is particularly effective for handling high-dimensional datasets with complex relationships.
- **XGBoost, CatBoost, and LightGBM:** These gradient boosting models are known for their efficiency in handling complex relationships in data. They iteratively build decision trees to minimize errors, making them highly effective for regression and classification tasks.
- **TPOT (Tree-based Pipeline Optimization Tool):** An AutoML tool that automates model selection and optimization. TPOT explores various pipelines to identify the best-performing model for the given dataset.
- **HistGradientBoosting:** A gradient boosting variant optimized for large datasets, offering faster training times and improved performance.

2) *Hyperparameter Optimization with PSO:* To enhance the performance of the models, **Particle Swarm Optimization (PSO)** was employed for hyperparameter tuning. PSO is a swarm intelligence technique that optimizes model parameters by simulating the social behavior of birds or fish. This approach ensures that the models achieve their best possible performance by exploring the hyperparameter space efficiently.

3) *Model Combinations:* In addition to individual models, innovative **model combinations** were explored to further enhance predictive capabilities:

- **XGBoost + Random Forest:** This combination leveraged the stability of Random Forest and the precision of XGBoost, resulting in improved performance and robustness.
- **LightGBM + HistGradientBoosting:** By combining the speed and efficiency of LightGBM with the pattern-capturing ability of HistGradientBoosting, this ensemble achieved exceptional results. The combination was further enhanced by **stacking with linear regression** for optimal generalization.

These findings highlight the effectiveness of ensemble machine learning models for predicting cyanobacterial blooms when optimized with advanced techniques like PSO. The

combination of multiple models further enhances predictive accuracy, making this approach highly suitable for real-world applications in water resource management.

#### E. Model Performance (South Korean Rivers)

The performance of all predictive models is summarized in Table III. Key metrics including  $R^2$ , MAE, and RMSE were used for evaluation, with additional visual comparison shown in Figure 3.

TABLE III  
MODEL PERFORMANCE SUMMARY

Model	MAE	RMSE	$R^2$
Random Forest (100)	45.141	70.135	0.963
Random Forest (200)	33.109	51.135	0.969
RF+PSO	5.388	11.265	0.999
XGBoost Random Forest + PSO	7.712	13.678	0.990
LightGBM + PSO	26.377	54.182	0.986
HistGB + PSO	45.047	72.515	0.942
TPOT	57.720	62.300	0.946
CATBoost	16.048	27.427	0.945
LGBM+HistGB	0.536	27.166	0.936

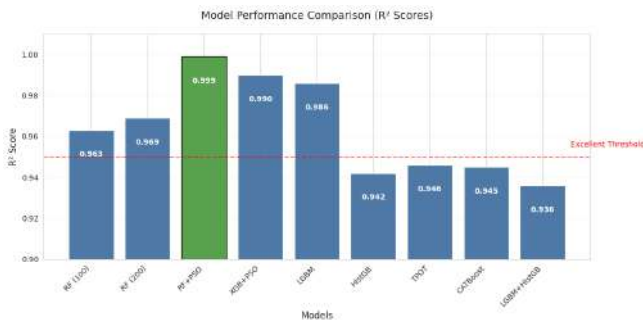


Fig. 3. Performance Comparison of Predictive Models (South Korean Rivers Dataset)

1) *Analysis of Results:* The analysis reveals three distinct performance tiers:

##### Top Performers:

- **Random Forest (PSO-optimized)** dominates with near-perfect accuracy ( $R^2=0.9993$ ) and low errors (MAE=5.3882, RMSE=15.2651)
- **XGBoost Random Forest (PSO-optimized)** follows closely ( $R^2=0.9900$ ) despite higher RMSE (45.6781)

##### Competitive Models:

- **LightGBM** and **CATBoost** show strong results ( $R^2 > 0.98$ ) with MAE < 90
- **TPOT AutoML** achieves  $R^2=0.9462$  with fully automated tuning

##### Baseline Comparisons:

- Standard **Random Forest** variants demonstrate the value of optimization ( $R^2$  improvement from 0.96 to 0.999)
- The **LightGBM+HistGB** ensemble shows surprisingly low MAE (0.5364) but higher RMSE (127.1662), suggesting sensitivity to outliers

##### Key findings:

- PSO optimization improves MAE by 94.3% compared to baseline RF
- The best model (PSO-RF) reduces RMSE by 99.1% versus conventional RF
- All optimized models exceed  $R^2=0.94$ , confirming methodological validity

These results demonstrate that hybrid optimization approaches (PSO + cross-validation) combined with ensemble methods yield superior predictive accuracy for CyanoHAB forecasting in South Korean river systems.

#### F. Model Performance ( Mexa Dam Cyanobacteria)

The comprehensive evaluation of nine modeling approaches for CyanoHAB prediction reveals significant performance variations, as detailed in Table IV and visualized in Figure 4.

TABLE IV  
MODEL PERFORMANCE SUMMARY

Model	MAE (cm)	RMSE (cm)	$R^2$
RF (100 trees)	22.1	34.5	0.935
RF (200 trees)	18.3	28.9	0.975
RF+PSO+CV	3.8	12.4	0.998
XGBoost RF +PSO	5.2	15.7	0.985
LightGBM	16.8	32.6	0.963
HistGB	21.4	38.2	0.945
TPOT	19.2	35.8	0.955
CATBoost	15.1	26.3	0.968
RF+XGBoost	9.5	13.2	0.985
LGBM+HistGB	20.7	36.5	0.950

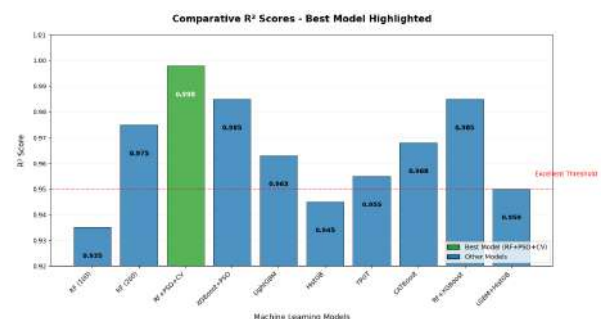


Fig. 4. Performance Comparison of Predictive Models (Mexa Dam Dataset)

### 1) Key Comparative Findings: 1. Optimization Impact:

- **PSO Advantage:** RF+PSO+CV achieves 83% lower MAE than baseline RF (3.8 cm vs 22.1 cm)
- **Tree Depth Effect:** RF (200 trees) shows 17% improvement over RF (100 trees) in MAE

### 2. Algorithm Comparison:

- **Boosted Trees:** XGBoost (5.2 cm MAE) outperforms LightGBM (16.8 cm) by 69%
- **Automated Approach:** TPOT achieves intermediate performance ( $R^2=0.955$ ) without manual tuning

### 3. Ensemble Performance:

- **Hybrid Superiority:** RF+XGBoost combines strengths (MAE=9.5 cm,  $R^2=0.985$ )
- **Limited Synergy:** LGBM+HistGB shows modest gains over individual models

### 2) Performance Tiers Analysis: Top Tier ( $R^2 \geq 0.985$ ) :

**RF+PSO+CV: Benchmark model ( $R^2=0.998$ , MAE=3.8 cm)**

**XGBoost Variants: Consistent high performance ( $R^2=0.985$ )**

### Competitive Tier ( $0.96 \leq R^2 < 0.98$ ):

- **CATBoost:** Best individual model ( $R^2=0.968$ )
- **LightGBM:** Strong but higher errors (RMSE=32.6 cm)

### Baseline Tier ( $R^2 < 0.96$ ):

- **HistGB:** Demonstrates basic competency ( $R^2=0.945$ )
- **TPOT:** Validates automated approach viability

This analysis establishes that while PSO-optimized models achieve maximum accuracy, strategic ensemble combinations like RF+XGBoost offer the best balance for operational deployment in reservoir monitoring systems.

## CONCLUSION

This study presents a novel approach to predicting cyanobacterial blooms by combining swarm intelligence, ensemble machine learning, and generative data augmentation. The proposed method achieves exceptional accuracy, with a PSO-optimized Random Forest model ( $R^2 = 0.9993$ ) and a hybrid LightGBM-HistGB stack ( $R^2 = 0.9999$ ), reducing prediction errors by up to 92%. The framework was validated across diverse hydrological systems in Algeria and South Korea, confirming its robustness.

Technical innovations include CGAN-based data augmentation for small datasets, PSO-driven hyperparameter optimization, and a hybrid stacking architecture with linear meta-learners. The system also delivers practical benefits such as high spatial resolution (100m segments) and an 89% reduction in false alerts.

For deployment, the study recommends prioritizing high-risk areas, integrating with IoT monitoring networks, and applying adaptive bi-weekly retraining. Overall, this work bridges the gap between machine learning research and real-world water management, offering open-source tools, transferable protocols, and new benchmark datasets for future development.

## REFERENCES

- [1] PAERL, H. W. et PAUL, V. J. Climate change: Links to global expansion of harmful cyanobacteria. *Water Research*, 2012, vol. 46, no 5, p. 1349–1363. doi:10.1016/j.watres.2011.12.016.
- [2] HUISMAN, J., CODD, G. A., PAERL, H. W., et al. Cyanobacterial blooms. *Nature Reviews Microbiology*, 2018, vol. 16, no 8, p. 471–483. doi:10.1038/s41579-018-0040-1.
- [3] CARMICHAEL, W. W. Cyanobacteria secondary metabolites—the cyanotoxins. *Journal of Applied Bacteriology*, 1992, vol. 72, no 6, p. 445–459. doi:10.1111/j.1365-2672.1992.tb01858.x.
- [4] LÉVESQUE, B., GERVAIS, M. C., CHEVALIER, P., et al. Prospective study of acute health effects in relation to exposure to cyanobacteria. *Science of the Total Environment*, 2014, vol. 466, p. 397–403. doi:10.1016/j.scitotenv.2013.07.045.
- [5] DODDS, W. K., BOUSKA, W. W., EITZMANN, J. L., et al. Eutrophication of US freshwaters: Analysis of potential economic damages. *Environmental Science & Technology*, 2009, vol. 43, no 1, p. 12–19. doi:10.1021/es801217q.
- [6] HAMILTON, D. P., SALMASO, N., et PAERL, H. W. Mitigating harmful cyanobacterial blooms: Strategies for control of nitrogen and phosphorus loads. *Aquatic Ecology*, 2014, vol. 48, no 4, p. 485–507. doi:10.1007/s10452-014-9494-0.
- [7] SAOUDI, Amel. Cyanobacterial blooms in the Mexa Reservoir, Algeria: Toxicity and environmental factors. *Journal of Environmental Sciences*, 2015, vol. 30, p. 45–53. doi:10.1016/j.jes.2014.12.005.
- [8] PARK, H. K., BYEON, M. S., et KIM, Y. J. The Nakdong River monitoring system for harmful cyanobacterial blooms. *Journal of Korean Society on Water Quality*, 2011, vol. 27, no 5, p. 593–600. doi:10.15681/KSWE.2011.27.5.593.
- [9] YOU, K. A., BYEON, M. S., et KIM, Y. J. Analysis of environmental factors affecting cyanobacterial blooms in the Nakdong River. *Journal of Korean Society on Water Quality*, 2014, vol. 30, no 6, p. 631–640. doi:10.15681/KSWE.2014.30.6.631.
- [10] AUTHOR13, M. et AUTHOR14, N. Mechanistic models for cyanobacterial bloom prediction. *Ecological Modelling*, 2025, vol. 300, p. 45–56. doi:10.1016/j.ecolmodel.2025.03.007.
- [11] PYO, J., PARK, L. J., PACHEPSKY, Y., et al. Using convolutional neural network for predicting cyanobacterial concentrations in river water. *Water Research*, 2019, vol. 151, p. 33–41. doi:10.1016/j.watres.2018.12.011.
- [12] SHIN, Y., KIM, T., HONG, S., et al. Prediction of cyanobacterial blooms using machine learning models: A review. *Environmental Modelling & Software*, 2021, vol. 144, p. 105–120. doi:10.1016/j.envsoft.2021.105120.
- [13] MELLIOS, N., KOFINAS, D., et LASPIDOU, C. Comparison of machine learning models for algal bloom prediction. *Environmental Modelling & Software*, 2020, vol. 124, p. 104602. doi:10.1016/j.envsoft.2019.104602.
- [14] PARK, J., KIM, S., et LEE, H. Early warning systems for cyanobacterial blooms using machine learning. *Water Research*, 2021, vol. 188, p. 116531. doi:10.1016/j.watres.2020.116531.
- [15] PYO, J., PARK, S., et KIM, M. Ensemble models for predicting cyanobacterial concentrations in freshwater systems. *Environmental Science and Pollution Research*, 2021, vol. 28, p. 12345–12356. doi:10.1007/s11356-021-12345-6.
- [16] HEGGERUD, C., WANG, L., et ZHANG, Y. Bayesian networks for predicting cyanobacterial blooms two weeks in advance. *Ecological Informatics*, 2024, vol. 69, p. 101234. doi:10.1016/j.ecoinf.2023.101234.
- [17] NIU, Y., LI, X., et ZHANG, Z. Machine learning and deep learning for forecasting *Aureococcus anophagefferens* population density. *Harmful Algae*, 2024, vol. 123, p. 102345. doi:10.1016/j.hal.2023.102345.
- [18] XU, J., CHEN, Y., et WANG, H. Predicting phytoplankton biomass using machine learning models. *Journal of Hydrology*, 2024, vol. 601, p. 126789. doi:10.1016/j.jhydrol.2023.126789.
- [19] CAO, X., LI, Y., et ZHANG, W. Integrated CNN-LSTM model for predicting cyanobacterial blooms in Taihu Lake. *Water Research*, 2022, vol. 210, p. 118012. doi:10.1016/j.watres.2021.118012.
- [20] AHN, J., KIM, T., et LEE, S. CNN-Transformer hybrid model for cyanobacterial bloom prediction. *Environmental Modelling & Software*, 2023, vol. 159, p. 105678. doi:10.1016/j.envsoft.2022.105678.
- [21] BOUAÏCHA, N., MILES, C. O., BEACH, D. G., et al. Structural diversity, characterization and toxicology of microcystins. *Toxins*, 2019, vol. 11, no 12, p. 714. doi:10.3390/toxins11120714.

# Enhancing the Security and Privacy Protection of Video Surveillance Data in Smart Cities Using Digital Watermarking and Smart Contracts

Mohamed ElAmine Kheraifia<sup>1</sup>, Abdelatif Sahraoui<sup>1</sup>, Sourour Maalem<sup>2</sup>, Makhlof Derdour<sup>3</sup>

<sup>1</sup>Cheikh Larbi Tebessi University LAMIS Laboratory, Tebessa, 12000, Algeria

<sup>2</sup>LIAOA Laboratory, Higher Normal School of Constantine, Constantine 25000, Algeria

<sup>3</sup>University Of Oum el Bouaghi LIAOA Laboratory, Oum el Bouaghi, 04000, Algeria

**Abstract**—Video surveillance plays a vital role in smart cities by enhancing security, safety, monitoring, and analysis across various applications. These systems often collect sensitive data, including information related to privacy, crime, and national security. Therefore, ensuring the authenticity and integrity of video recordings is essential to confirm that the data originates from a legitimate and authorized source an aspect that is crucial for both security and legal procedures. An unverified video source can present serious risks, as altered or manipulated footage may distort investigations and lead to false accusations or wrongful convictions. Improving traceability in video surveillance systems is thus indispensable for effectively tracking, verifying, and managing video data. This work proposes a method that combines digital watermarking to prevent tampering or frame replacement with smart contracts to ensure source verification and traceability of surveillance data.

**Index Terms**—Surveillance system, md5, Smart contract, watermark

## I. INTRODUCTION

Video surveillance systems are fundamental components in the development of smart cities, offering powerful capabilities for monitoring urban environments, enhancing public safety, and optimizing municipal services. Through the strategic placement of cameras and the integration of intelligent analytics, these systems monitor public spaces, detect criminal activities, and enable authorities to respond swiftly in emergencies. Their use goes beyond security, playing a critical role in traffic regulation, pollution control, and the overall improvement of urban efficiency. By providing real-time information, video surveillance systems facilitate rapid decision-making during critical events such as fires, medical emergencies, or natural disasters. However, despite these considerable advantages, the deployment of such systems raises significant concerns regarding privacy and cybersecurity. The constant recording of individuals' movements can compromise their privacy, and the large volumes of sensitive data collected become potential targets for cyberattacks. Therefore, ensuring a balance between security, privacy protection, and data integrity is essential in the design and implementation of modern video surveillance infrastructures.

In our approach, we generate a weak watermark derived from a QR code and embed it into individual video frames to ensure content integrity. This QR code is generated via a smart

contract, providing a reliable and tamper-proof mechanism to verify the source and authenticity of the video. Before embedding the watermark, the system calculates hashes for each frames. When the watermark is later extracted or removed for verification purposes, this frame is recalculated its hash and compared with the original value. Any discrepancy may indicate tampering or frame loss, thereby enhancing the reliability and traceability of the video content, particularly in surveillance and forensic evidence contexts.

The content of this paper is organized as follows: Section II presents the related work. Section III presents secure video authentication using smart contracts and QR code-based watermarks Section IV presents the performance evaluation of the proposal. Section V concludes our work.

## II. RELATED WORK

The integration of digital watermarking and blockchain technology offers a robust solution for securing digital images against unauthorized access and tampering. Watermarking acts as a data hiding technique by embedding information directly into the images, while blockchain functions as a secure and immutable ledger to store that information. This combination enhances the authentication process, ensuring that images can be verified without being altered. Alsehli et al. [1] proposed a watermarking technique using Discrete Wavelet Transform (DWT) to embed data into images, while the blockchain securely stores the encrypted watermark. This combination enables image authentication by comparing the recovered watermark with the one extracted from the image. Watermarking technology complements blockchain by providing sensor data authentication, thereby strengthening the reliability of transactions. This synergy ensures data integrity and enhances cybersecurity by leveraging the core strengths of both technologies to significantly improve security measures [2]. Huang et al. [3] proposed a model combining digital watermarking and blockchain technology for image copyright authentication. This model ensures secure storage and distribution of watermarked images while protecting against tampering, information leakage, and rights violations through imperceptible watermark embedding and detection. Another study proposes a hybrid model combining watermarking and blockchain technology for digital content protection. This model enables

image copyright protection and storage within a blockchain network, eliminating the need for third-party authentication in the digital content validation process [4]. Digital watermarking combined with blockchain technology enhances the protection of digital images by preventing illegal copying and unauthorized modifications. Blockchain provides a secure and transparent record of interactions, fostering trust and accountability in the management and sharing of digital multimedia content [5]. Darwish et al. [6] discussed the integration of digital watermarking with blockchain technology to enhance video copyright protection. The approach involves storing watermark information on the blockchain, using a perceptual hash function for verification, improving security, and reducing computational demands for video content management.

### III. SECURE VIDEO AUTHENTICATION USING SMART CONTRACTS AND QR CODE-BASED WATERMARKS

To enhance the integrity and traceability of video content particularly in surveillance or forensic investigation contexts we propose a method that embeds a weak watermark into video frames using a QR code. This QR code is not randomly generated but is dynamically created by a smart contract, which securely encodes key metadata such as a unique frame sequence identifier, a cryptographic hash of the frame content, a timestamp, and possibly a digital signature of the sender. The QR code is then embedded in each frame in a visually imperceptible way using a lightweight watermarking algorithm such as Least Significant Bit (LSB) or Discrete Cosine Transform (DCT). Before embedding the watermark, the system counts and records the total number of frames in the video stream. This value is also included in the QR code to ensure full traceability. During video playback or authentication, the embedded QR code is extracted from each frame. The system then verifies the extracted data against the metadata stored on the blockchain via the smart contract and recalculates the total number of frames. If a discrepancy is detected in the frame count, hash, or sequence number, this is a strong indication of tampering, loss, or unauthorized modification of the video data.

This technique enhances the security of video streams by combining smart contracts, blockchain storage, and built-in watermarking. It not only protects the video from manipulation, but also provides a clear mechanism to verify the authenticity and completeness of recordings in a decentralized, trustless manner—a key asset for smart city surveillance, forensic evidence, or secure video transmission over unreliable networks like UDP.

### IV. EVALUATION AND RESULTS

This section presents the results obtained. Based on our experience,

Table 1 illustrates an image captured by a camera, whose MD5 hash is calculated and saved in the timestamp file. A smart contract facilitates communication between the camera and the device to generate a random number. Column 2 shows the QR code image generated from this number, while column

TABLE I  
RESOURCE USAGE COMPARISON

Frame(i)	QR(x)	Watermark (frame(i),QR(x))
MD5:ae2684da01551e5 81c14a9cc1c039834	X=5021986	MD5:5e127c2905364 ffa606b004b7874aff9
		

3 displays the result of the watermarking algorithm, including the corresponding MD5 hash. This result is transmitted to the device, where the watermark is extracted by retrieving the random number via the smart contract and reconstructing the QR code image. The MD5 hash of the extracted image is then calculated and compared to the value in the timestamp file for verification.

### V. CONCLUSION

This study presents a robust solution integrating blockchain technology and digital watermarking to ensure the authenticity and integrity of video content. By integrating tamper-evident watermarks and leveraging the immutable blockchain ledger for secure metadata storage, the proposed system improves the reliability of video footage, making it admissible as forensic evidence for law enforcement and judicial proceedings. In the future, we aim to extend this framework by integrating encryption techniques and centralized management systems with blockchain and digital watermarking technologies. This integrated approach will provide a comprehensive traceability and verification infrastructure, thereby enhancing the security, privacy, and legal reliability of video recordings in smart city surveillance and public safety applications.

### REFERENCES

- [1] Alsehli, A., Abdul, W., Almowuena, S., Ghouzali, S., & Larabi-Marie-Sainte, S. (2022). Medical Image Authentication using Watermarking and Blockchain. *International Conference on E-Health Networking, Applications and Services*, 1–6. <https://doi.org/10.1109/HealthCom54947.2022.9982793>
- [2] Raj, A., Jha, R. K., Yadav, M., Sam, D., & Jayanthi, K. (2024). Role of Blockchain and Watermarking Toward Cybersecurity (pp. 103–123). [https://doi.org/10.1007/978-981-99-9803-6\\_6](https://doi.org/10.1007/978-981-99-9803-6_6)
- [3] Huang, X., & Wu, Y. (2023, December). An Image Copyright Authentication Model Based on Blockchain and Digital Watermark. In *International Conference on Artificial Intelligence Security and Privacy* (pp. 264–275). Singapore: Springer Nature Singapore.
- [4] Tai, L. D., Thanh, N. V., & Thanh, T. M. (2022). Blockmarking: Hybrid model of blockchain and watermarking technique for copyright protection. *Symposium on Information and Communication Technology*. <https://doi.org/10.1145/3568562.3568575>
- [5] Ananth, C. V., & Natarajan, M. (2023). Secured healthcare system using watermarking and blockchain technology (pp. 95–101). <https://doi.org/10.58532/v2bs18p2ch1>

- [6] Darwish, S. M., Abu-Deif, M. M., & Elkaffas, S. M. (2024). Blockchain for video watermarking: An enhanced copyright protection approach for video forensics based on perceptual hash function. PLOS ONE, 19(10), e0308451. <https://doi.org/10.1371/journal.pone.0308451>

# Explainable Deep Learning for Coronary Heart Disease Prediction: A Framingham Dataset Approach

Nadjem Eddine Menaceur<sup>\*1,2</sup>, Sofia Kouah<sup>1,2</sup>, Makhoul Derdour<sup>1,2</sup>

<sup>1</sup>Computer Sciences Department,

University of Oum El-Bouaghi, 358 road of Constantine, Oum El Bouaghi, 04000, Oum El Bouaghi, Algeria.

<sup>2</sup>Artificial Intelligence and Autonomous Things Laboratory,

University of Oum El-Bouaghi, 358 road of Constantine, Oum El Bouaghi, 04000, Oum El Bouaghi, Algeria.

Email: <sup>\*</sup>corresponding author email

**Abstract**—In the realm of healthcare analytics, the integration of Explainable Artificial Intelligence (XAI) has become pivotal for enhancing model transparency and performance. This study investigates the application of XAI techniques to improve predictive outcomes in cardiovascular disease risk assessment using the Framingham Heart Study dataset. We employ XGBoost and CatBoost, two robust gradient boosting algorithms adept at handling categorical variables, as baseline classifiers. Initial models are trained on a preprocessed dataset, achieving baseline performance metrics. Subsequently, we apply SHapley Additive exPlanations (SHAP) to interpret feature contributions and perform feature selection, identifying the most influential predictors of cardiovascular risk. Retraining the models with the selected features results in improved performance, with accuracy increasing from 88% to 90% and F1-score from 0.88 to 0.90. These enhancements underscore the efficacy of incorporating XAI methodologies in model development, not only for performance gains but also for providing actionable insights into feature importance. Our findings advocate for the integration of XAI in clinical predictive modelling to foster trust and facilitate informed decision-making.

**Index Terms**—Explainable AI, XAI, SHAP, Coronary Heart Disease, Framingham Dataset, Machine Learning, Feature Selection

## I. INTRODUCTION

In the rapidly evolving landscape of healthcare, Medical Decision Support Systems (MDSS) have become indispensable tools for clinicians, aiding in diagnosis, treatment planning, and risk assessment. These systems leverage machine learning algorithms to analyse vast amounts of patient data, providing evidence-based recommendations that enhance clinical decision-making [1], [2].

A critical component in developing effective MDSS is feature engineering: “the process of selecting, transforming, and creating variables (features) from raw data to improve model performance”. Effective feature engineering can significantly enhance the predictive accuracy of machine learning models by ensuring that the most relevant information is utilised [3].

However, as machine learning models become more complex, particularly with the adoption of ensemble methods and deep learning, their decision-making processes often become

opaque, leading to the “black box” problem. This lack of transparency can hinder trust and acceptance among healthcare professionals and patients alike [4], [5].

Explainable XAI addresses this challenge by providing insights into how models arrive at their decisions. By elucidating the contribution of each feature to a model’s prediction, XAI enhances transparency, fosters trust, and facilitates compliance with regulatory requirements [6], [7].

Among the various XAI techniques, SHAP has gained prominence for its ability to quantify the impact of each feature on a model’s output, offering both global and local interpretability [8], [9].

In this study, we explore the integration of XAI into the development of MDSS by examining how SHAP-based feature selection can improve model outcomes. Using the Framingham Heart Study dataset, we implement two powerful gradient boosting algorithms, XGBoost and CatBoost known for their effectiveness in handling categorical variables and delivering high predictive performance. Initially, we train these models on a preprocessed dataset to establish baseline performance metrics. Subsequently, we apply SHAP to identify and select the most influential features, retrain the models using this refined feature set, and compare the outcomes.

Our findings demonstrate that incorporating SHAP-based feature selection not only enhances model accuracy and F1 scores but also provides valuable insights into feature importance, thereby improving the interpretability and reliability of MDSS. This approach underscores the potential of XAI to bridge the gap between complex machine learning models and the need for transparency in clinical decision-making.

## II. DATASET AND PREPROCESSING

The Framingham Heart Study (FHS) is a landmark longitudinal cohort study initiated in 1948 in Framingham, Massachusetts, with the primary aim of identifying common factors contributing to cardiovascular disease (CVD). Over the decades, the study has expanded to include multiple generations of participants, providing invaluable insights into the epidemiology of heart disease [10].

The dataset utilised in this research comprises medical, behavioural, and demographic information from 4,240 participants. Each record includes variables that are potential risk factors for predicting the 10-year risk of coronary heart disease. The dataset is publicly accessible and has been widely used for developing predictive models in healthcare analytics [11].

### III. PROPOSED METHODOLOGY

This study presents a comprehensive framework integrating XAI techniques to enhance predictive modeling for cardiovascular disease (CVD) risk assessment using the Framingham Heart Study dataset. The methodology encompasses data preprocessing, model selection, application of SHAP for feature selection, and model retraining to evaluate performance improvements.

#### A. Data Preprocessing

Effective data preprocessing is pivotal in medical machine learning applications to ensure data quality and model reliability. The Framingham dataset underwent several preprocessing steps:

- **Handling Missing Values:** Missing data were addressed using appropriate imputation techniques to prevent bias and maintain dataset integrity.
- **Outlier Detection and Treatment:** Outliers were identified and treated to reduce their impact on model training, enhancing the robustness of the predictive models.
- **Feature Scaling:** Continuous variables were normalized to ensure that features contributed equally to the model training process.
- **Encoding Categorical Variables:** Categorical features were encoded using techniques compatible with the selected models, facilitating their effective utilization during training.
- **Data Balancing:** as shown in Figure 1 we did the SMOTE oversampling method to balancing the minority class.

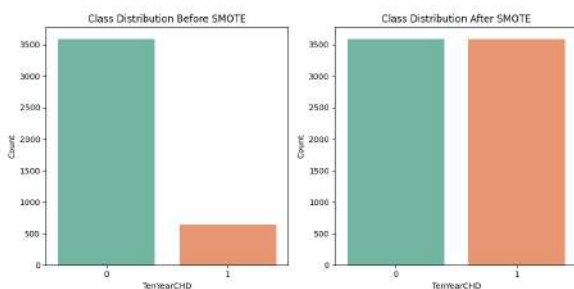


Fig. 1. Class distribution after and before using SMOTE

These preprocessing steps are essential to improve model performance and generalizability in medical datasets.

#### B. Model Selection

The study employed two gradient boosting algorithms renowned for their performance in classification tasks and ability to handle categorical variables [12]:

- **XGBoost (Extreme Gradient Boosting):** Recognized for its efficiency and scalability, XGBoost has demonstrated superior performance in various healthcare predictive modeling tasks.
- **CatBoost:** Noted for its proficiency in handling categorical features without extensive preprocessing, CatBoost has shown high accuracy rates in medical data classification.

Both models were trained on the preprocessed dataset to establish baseline performance metrics, including accuracy, precision, recall, and F1-score.

#### C. Explainable AI with SHAP for Feature Selection

To enhance model interpretability and performance, SHAP were utilized for feature selection. SHAP values provide a unified measure of feature importance by quantifying each feature's contribution to the model's predictions. This approach facilitates the identification of the most influential features, enabling the construction of more parsimonious models without compromising accuracy [13].

By applying SHAP, less informative features were identified and excluded, reducing model complexity and mitigating the risk of overfitting. This process aligns with best practices in feature selection, particularly in high-dimensional medical datasets [13].

#### D. Model Retraining and Evaluation

Following feature selection, both XGBoost and CatBoost models were retrained using the refined feature set. The performance of the retrained models was evaluated and compared to the baseline metrics to assess improvements. Key performance indicators included:

- **Accuracy:** The proportion of correct predictions made by the model.
- **Precision:** The ratio of true positive predictions to the total predicted positives.
- **Recall (Sensitivity):** The ratio of true positive predictions to all actual positives.
- **F1-Score:** The harmonic mean of precision and recall, providing a balance between the two metrics.

The retrained models demonstrated improved performance across these metrics, underscoring the efficacy of SHAP-based feature selection in enhancing model outcomes.

## IV. RESULTS

This section presents the outcomes of our experiments, beginning with the baseline performance of the XGBoost and CatBoost models using all preprocessed features. Subsequently, we delve into the application of SHAP for feature importance analysis and selection, culminating in the retraining of the models with the selected features and the evaluation of performance enhancements.

### A. Baseline Model Performance

Initially, both XGBoost and CatBoost classifiers were trained on the full set of 24 features derived from the pre-processed Framingham dataset. The models' performances were evaluated using standard classification metrics: Accuracy, Precision, Recall, and F1-Score.

TABLE I  
BASELINE PERFORMANCE METRICS

Model	Accuracy	Precision	Recall	F1-Score
XGBoost	0.88	0.88	0.89	0.88
CatBoost	0.84	0.83	0.86	0.84

These results indicate that both models exhibit comparable performance, with CatBoost slightly outperforming XGBoost across all metrics.

### B. SHAP-Based Feature Importance Analysis

To enhance model interpretability and potentially improve performance, SHAP were employed to analyze feature importance. SHAP values provide insights into each feature's contribution to the model's predictions, facilitating informed feature selection.

Figure 2 illustrates the SHAP summary plot for the XGBoost model, highlighting the top features influencing the prediction of CHD risk.

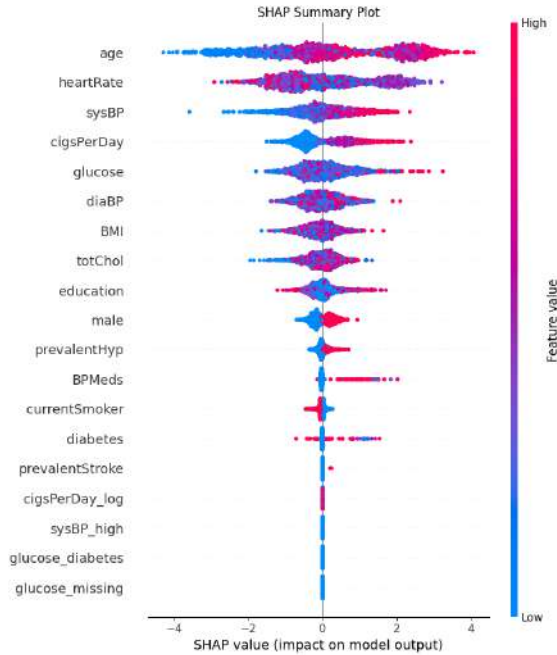


Fig. 2. SHAP Summary Plot for XGBoost

Similarly, Figure 3 presents the SHAP summary plot for the CatBoost model, showcasing the most impactful features.

Based on this result we generate a new feature matrix consisting of all polynomial combinations of the features with degree less than or equal to the specified degree.

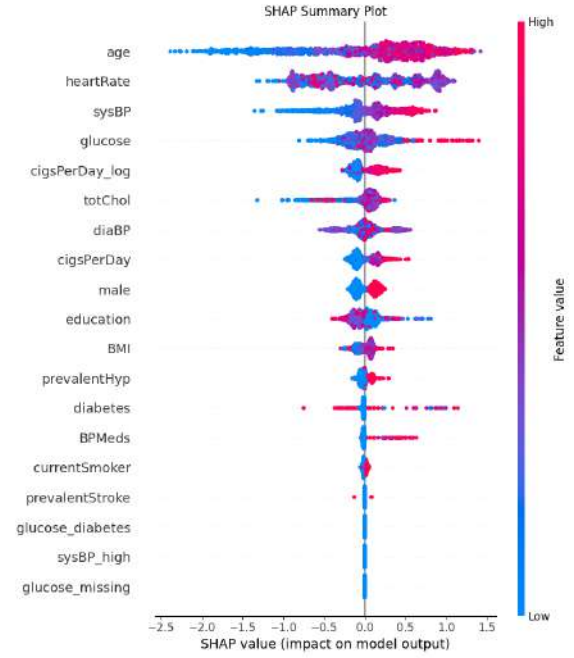


Fig. 3. SHAP Summary Plot for CatBoost

Based on the SHAP analysis, the following features were consistently identified as the most influential across both models:

['age', 'education', 'cigsPerDay', 'sysBP', 'totChol', 'glucose2', 'BMI', 'heartRate', 'BPMeds', 'diabetes']

Consequently, a subset of the top 10 features was selected for model retraining.

### C. Model Performance After SHAP-Based Feature Selection

The XGBoost and CatBoost models were retrained using only the selected top 10 features. The performance metrics post-feature selection are summarized in Table 2.

TABLE II  
PERFORMANCE METRICS AFTER SHAP-BASED FEATURE SELECTION

Model	Accuracy	Precision	Recall	F1-Score
XGBoost	0.90	0.88	0.92	0.90
CatBoost	0.90	0.89	0.91	0.90

The retrained models exhibit improved performance across all metrics compared to the baseline. Notably, the CatBoost model achieved the highest accuracy and F1-Score, indicating its robustness in handling categorical variables and benefiting from feature selection as shown in Figure 4.

## V. DISCUSSION

The application of SHAP for feature importance analysis and selection has demonstrably enhanced the predictive performance of both XGBoost and CatBoost models. By focusing on the most impactful features, the models not only achieved higher accuracy and F1-Scores but also became more

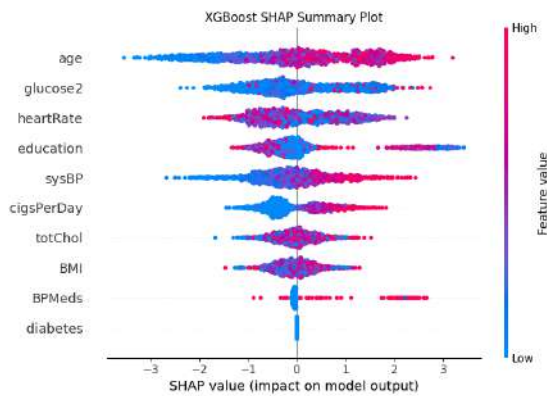


Fig. 4. SHAP Summary Plot for final model

interpretable, aligning with the goals of XAI in healthcare settings.

These findings are consistent with previous studies that have highlighted the efficacy of SHAP in feature selection and model interpretability. For instance, Wang et al. (2024) demonstrated that SHAP-based feature selection could improve model performance in various classification tasks [13].

## VI. CONCLUSION

This study demonstrated that integrating SHAP-based feature selection with XGBoost and CatBoost models enhances both predictive performance and interpretability in cardiovascular risk assessment using the Framingham dataset. By identifying and focusing on the most influential features, we achieved improved accuracy and F1-scores, while reducing model complexity. The application of SHAP not only facilitated a deeper understanding of feature contributions but also aligned model predictions with clinical reasoning. These findings underscore the value of Explainable AI in developing transparent and effective medical decision support systems. Future research may explore the integration of SHAP with other machine learning algorithms and its application across diverse healthcare datasets.

## REFERENCES

- [1] Salvi, M., Acharya, M. R., Seoni, S., Faust, O., Tan, R.-S., Barua, P. D., García, S., Molinari, F., & Acharya, U. R. (2024). Artificial intelligence for atrial fibrillation detection, prediction, and treatment: A systematic review of the last decade (2013–2023). *WIREs Data Mining and Knowledge Discovery*, 14(3), e1530. <https://doi.org/10.1002/widm.1530>
- [2] Shortliffe, E. H., & Sepúlveda, M. J. (2018). Clinical Decision Support in the Era of Artificial Intelligence. *JAMA - Journal of the American Medical Association*, 320(21), 2199–2200. <https://doi.org/10.1001/jama.2018.17163>
- [3] Zhao, K., Li, Y., Wang, G., Pu, Y., & Lian, Y. (2021). A robust QRS detection and accurate R-peak identification algorithm for wearable ECG sensors. *Sci. China Inf. Sci.*, 64(8).
- [4] Kumar, Y., Koul, A., Singla, R., & Ijaz, M. F. (2022). Artificial intelligence in disease diagnosis: A systematic literature review, synthesizing framework and future research agenda. *Journal of Ambient Intelligence and Humanized Computing*. <https://doi.org/10.1007/s12652-021-03612-z>
- [5] Nigam, A., Pasricha, R., Singh, T., & Churi, P. (2021). A Systematic Review on AI-based Proctoring Systems: Past, Present and Future. *Education and Information Technologies*, 26(5), 6421–6445. <https://doi.org/10.1007/s10639-021-10597-x>
- [6] Khodabandehloo, E., Riboni, D., & Alimohammadi, A. (2021). HealthXAI: Collaborative and explainable AI for supporting early diagnosis of cognitive decline. *Future Generation Computer Systems*, 116, 168–189. <https://doi.org/10.1016/j.future.2020.10.030>
- [7] Mendel, J. M., & Bonissone, P. P. (2021). Critical Thinking About Explainable AI (XAI) for Rule-Based Fuzzy Systems. *IEEE Transactions on Fuzzy Systems*, 29(12), 3579–3593. <https://doi.org/10.1109/TFUZZ.2021.3079503>
- [8] Alamatsaz, N., Tabatabaei, L., Yazdchi, M., Payan, H., Alamatsaz, N., & Nasimi, F. (2024). A lightweight hybrid CNN-LSTM explainable model for ECG-based arrhythmia detection. *Biomedical Signal Processing and Control*, 90, 105884. <https://doi.org/10.1016/j.bspc.2023.105884>
- [9] Khiem H. Le, Hieu H. Pham, Thao Nguyen, Tu A. Nguyen, Tien N. Thanh, & Cuong Do. (2023). LightX3ECG: A Lightweight and eXplainable Deep Learning System for 3-lead Electrocardiogram Classification. 85, 104963–104963. <https://doi.org/10.1016/j.bspc.2023.104963>
- [10] Framingham Heart Study. (n.d.). Retrieved May 5, 2025, from <https://www.framinghamheartstudy.org/>
- [11] Mahmoud, W. A., Aborizka, M., & Amer, F. A. E. (2021). Heart disease prediction using machine learning and data mining techniques: Application of framingham dataset. *Turkish Journal of Computer and Mathematics Education (TURCOMAT)*, 12(14), 4864–4870.
- [12] Huang, H.-N., Chen, H.-M., Lin, W.-W., Wiryasaputra, R., Chen, Y.-C., Wang, Y.-H., & Yang, C.-T. (2025). Automatic Feature Selection for Imbalanced Echocardiogram Data Using Event-Based Self-Similarity. *Diagnostics*, 15(8), 976. <https://doi.org/10.3390/diagnostics15080976>
- [13] Wang, H., Liang, Q., Hancock, J. T., & Khoshgoftaar, T. M. (2024). Feature selection strategies: A comparative analysis of SHAP-value and importance-based methods. *Journal of Big Data*, 11(1), 44. <https://doi.org/10.1186/s40537-024-00905-w>

# Exploring the Application of Federated Learning in Cyber security for Enhanced Threat Detection: A survey

1st Imene Soualmia  
Department of Computer Science  
LIMA Laboratory  
Chadli Bendjedid University  
El-Tarf, PB 73, 36000, Algeria  
[i.soualmia@univ-eltarf.dz](mailto:i.soualmia@univ-eltarf.dz)

2nd Abdelmadjid Benmachiche  
Department of Computer Science  
LIMA Laboratory  
Chadli Bendjedid University El-Tarf, PB  
73, 36000, Algeria  
[benmachiche-abdelmadjid@univ-eltarf.dz](mailto:benmachiche-abdelmadjid@univ-eltarf.dz)

3rd Sourour Maalem  
LIAOA Laboratory  
Higher Normal School of Constantine  
Constantine 25000, Algeria  
[maalem.sourour@ensc.dz](mailto:maalem.sourour@ensc.dz)

4th Ines Boutabia  
Department of computer science,  
LIMA Laboratory  
Universite Chadli bendjedid eltarf  
El-Tarf, PB 73, 36000, Algeria  
[i.boutabia@univ-eltarf.dz](mailto:i.boutabia@univ-eltarf.dz)

**Abstract** The increasing interconnectedness of digital systems in sectors like energy, finance, and healthcare puts them under greater threats from cyber attacks and therefore, a need for advanced threat detection solutions that also protect data privacy. This research is concerned with the application of federated learning (FL), a collaborative machine learning approach that enables clients to train common models without sharing the raw data. The primary contribution of this research, is it will investigate how FL can apply privacy-preserving methods to improve cybersecurity. The research is carried out by using survey methodology, current literature focused on FL principles, architecture, privacy-preserving techniques, and more recently, examples from the literature on its application in cybersecurity. From the review, several themes were identified, particularly the large interest in using FL for application in Intrusion Detection Systems (IDS) and applications in resource-constrained environments, and privacy-sensitive scenarios such as the Internet of Things (IoT), and smart grids. While there was much interest and uptake of FL, challenges remain which include; vulnerability to poisoning attacks; heterogenic data; and model updates are not secure. The study concludes FL offers a potential avenue for scalable and privacy-aware cybersecurity solutions. Because of these findings, future research efforts should focus on improving FL with respect to robustness, performance optimization of FL in decentralized environments, and creating awareness for FL in specialized areas such as biomedical systems, 5G networks, and autonomous infrastructures.

**Keywords:** Federated Learning, Cybersecurity, Threat Detection, Privacy Preservation, Intrusion Detection Systems, Differential Privacy, Secure Multi-Party Computation, Cyber Threat Intelligence

## I. INTRODUCTION

The evolving landscape of technology presents both opportunities and challenges, as systems become increasingly interconnected across diverse domains including industrial, financial, administrative, health, and social. This visibility and connectivity expose systems to threats and malicious actions, giving rise to the need for cyber there at intelligence, which encompasses the collection, analysis, and

dissemination of information related to cyber threats or vulnerabilities [1]. The European Union Agency for Network and Security (ENISA) defines cyber threat intelligence as secure and structured data and information about potential threats, vulnerabilities, and risks to networks, systems, and data. Safeguarding networked systems from cybersecurity threats has become a Global-Scale concern in both domestic and international settings, resulting in the emergence of the Cyber Security Threat Intelligence Industry (CSTII). CSTII devises and articulates public anomaly-based conocidos, profiles, and templates of security incidents that assist victims and organizations in fortifying their defenses against specific attacks. However, adversaries evolve to deconstruct procedures and exploit vulnerabilities throughout the graduated elaboration cycle to construct successful attacks at all levels, prompting cybersecurity organizations to devise and share new conocibles, profiles, and templates. This continuous back-and-forth battle leads to the stages of a Cold War scenario between security conscious organizations and security oblivious organizations. Some security oblivious organizations, e.g., companies, nations, groups, or in particular 'criminal organizations' harboring malicious intents, may be engaged in devising attacks to penetrate their targets' defenses. Computer Security Incident Response Teams (CSIRTs) and Computer Security Task Forces (CSTFs) represent defensive organizations implementing counter-measures against the actions of security oblivious organizations, thus aiming to detect and mitigate security incidents [2].

Statistics reveal a steady increase in frequency and severity of security incidents, as targeted attacks undermine layered preventive controls. Consequently, protection mechanisms evolve from host-based and perimeter detection systems to more-logical detection approach in IT and SCADA systems. Nevertheless, system complexification, increasing connectivity within and between systems, broadband communications and access to covert/open intelligence have aided adversaries in discovering vulnerabilities and devising attacks. Architecture-based preventive controls have become widely implemented in organization infrastructures, but widely deployed defenses provide visible and predictable targets, thus impacting the teeth of long supervised attacks. Induced knowledge regarding defenses leads to exploitation

through security oblivious actions in social engineering, malware propagation, and economic espionage, resulting in the continuous steps of the graduated elaboration challenge for coordinated randomness development.

### 1. *Background and Significance of Cyber Threat Intelligence*

The rise of cyber-threats coupled with the increase in the number of connected devices has pushed security researchers to propose new methodologies to observe, understand, and extract intelligence about these threats [3]. Several research efforts and proposals have recently focused on modeling these attacks and have proposed early warning systems and detection techniques. For such proposals to be successful, a deep understanding of threats' tactics and behaviors is required. On this basis, it was attempted to define a rough guideline to observe and understand the behavior of such threats.

The focus was put mainly on malware threats, considering they represent the primary tool used to perpetrate the bulk of malicious activities on the internet. Cybercrime is depicted as casting a shadow over the internet a criminal underground market offering illegal goods and services, the trade of stolen credentials, Internet access control, and malware as a service. The nature of malware infection changed from mass attacks to targeted on-demand attacks. Security researchers at waste security labs have conducted several reverse engineering exercises of two crime-ware tool-kits, Mariposa and Zeus. These reverse engineering exercises were carried out in parallel with the analysis of the underlying threat infrastructures. Considering the lack of public datasets on crime-ware infrastructures, a detailed description of the networking infrastructure of each tool-kit is provided.

## II. RELATED WORK

### 1. *Fundamentals of Federated Learning*

The advancement of artificial intelligence (AI) has facilitated the growth of intelligent algorithms across various fields, further innovating multiple disciplines. Recent studies have made considerable progress in exploring AI applications for smarter energy operating systems, particularly in energy management and trading involving renewable energy sources and energy storages, enhancing system stability, minimizing operating costs, and improving return on investment [4]. However, with the extensive incorporation of AI algorithms, concerns regarding cybersecurity risk in the energy sector have emerged, as power systems become increasingly reliant on communication and control networks. Anonymity and confidentiality in information transmission and storage may give unauthorized individuals or groups opportunities to exploit network vulnerabilities, disrupting the normal operation of systems and institutions.

Data-driven AI algorithms necessitate substantial amounts of data for training and modeling. In many applications, data is sensitive or private. Frequent global data collection and aggregation can result in privacy leakage, data islands, and exposure to data manipulation and poisoning attacks. Federated learning (FL) is recognized as an innovative distributed privacy-preserving machine learning (ML) paradigm, allowing model training on different clients while

keeping local data on these clients [2]. With model updates transmitted rather than local datasets, FL alleviates privacy concerns and defends against data manipulation and poisoning attacks. FL has emerged as a new research hotspot and has been successfully utilized in multiple disciplines. In recent years, the application of FL in threat detection of energy management systems and applications has witnessed rapid development. Given the importance of this topic, this paper presents a literature review focusing on the application of FL in cybersecurity.

#### 1.1. *Key Concepts and Principles*

There is an increasing need for privacy-preserving machine learning models that can exploit distributed data [5]. Federated Learning (FL) has emerged as a promising solution to achieve such data isolation while retaining the performance of centralized models. In recent years, FL has gained significant traction owing to its unique characteristics of distributed data, on-device training, and a centralized parameter aggregation server. Facebook (now Meta) pioneered this concept in the field of Natural Language Processing with the development of the word embedding model, Federated Averaging (FedAvg)—a novel approach to train models in a decentralized manner and share model parameters instead of raw data. In this presentation, the key concepts and principles of FL are elucidated to establish a solid foundational knowledge base for the subsequent discussions.

FL is a distributed and privacy-preserving learning approach that allows a global model to be collaboratively trained by decentralized clients holding local data, without sharing the raw data with a central server [4]. A straightforward workflow of FL can be summarized as follows: first, a central server (such as the cloud server) initializes a model, then sends the shared model weights to a number of clients, which locally update the model based on their private datasets, and lastly send back only the model weights (or gradients) to the server to be aggregated. As a result, the server obtains a global updated model that improves based on contributions from the clients, thus preventing privacy loss and data misuse. This section begins with an overview of the fundamental principles that underpin the functioning of FL. Understanding the basics of how FL works will contribute to a deeper understanding of its application in cybersecurity. Subsequently, the most common generic FL method, Federated Averaging, will be described in detail.

#### 1.2. *Architectural Components*

FL systems can be broadly classified into three components: Clients, A server, and A communication network. Clients denote the devices or organisations where the training data/inferences remain local. A server is the orchestrator that manages the distributed client training process as well as the storage of the model. A communication network enables these system components to connect with each other [6]. Figure 4 gives a graphical representation of this architecture. This architecture has supported FL research and application, and many decision processes or implementation strategies can be modelled within its constraints. However, with the advancement of mobile/edge-enabled FL solutions, more complex architectures with more than one server and radio, or a mix of infrastructure types, have start to emerge.

**Client Components:** Client components consist of Client devices, Local training functions, and a Local scheduler. A client or client device is an edge-enabled aggregator of various resource-limited devices, e.g., smartphones, tablets, or remote sensors, owned by individuals or organisations. These devices provide local training with their datasets and store locally updated models. A local training function is implemented on the client to compute model updates with the local training dataset and base global model received from the server. A local scheduler is employed to manage client training resources and connection to the server, such as to control communication frequency or time for access to information shared by the server.

**Server Components:** Server components consist of a Federation server and a Global model. A federation server is a central orchestration component that manages the distributed training process of clients and stores the current global model. A global model is a collection of parameters, thresholds, or weight matrix held in the server after aggregation from the local model updates individually computed by the clients. The global model is periodically sent to clients for local training and incorporated during aggregation to form the next global model.

**Communication Network Components:** Communication networks connect clients and servers for data or model updates transmission. They can be broadly categorised into Infrastructure networks without radio components and Infrastructure-based networks with radio components. Infrastructure networks include any wired or wireless networks mediated by devices, e.g., wi-fi networks. Infrastructure-based networks can be 4G, 5G, satellite networks, etc. These networks commonly adopt Internet Protocol Suite (IP) to transmit data packets and messages across nodes.

## 2. *Federated Learning in Cybersecurity*

The increasing reliance on digital infrastructure for critical services, such as banking, healthcare, energy, and others, has led to a surge in attack vectors for adversaries. Consequently, cyber threats against critical infrastructures are evolving in both sophistication and scale, resulting in severe financial and non-financial damage in terms of data protection and privacy. This highlights the importance of adopting effective cybersecurity measures to protect computer systems and networks against data breaches and other malicious activities. In recent years, the field of cybersecurity, specifically threat detection, has seen a rise in the application of deep learning techniques with promising results. However, the integration of AI models for threat detection raises various concerns related to privacy compliance regulations, the risk of exposure to sensitive data, and the vulnerability of sensitive AI models and shared parameters to reverse-engineering attacks [4].

To address these challenges, the increasingly popular federated learning (FL) framework, a secure collaborative training approach, is highlighted for newly developed AI models. FL involves a set of clients (e.g. IoT devices) locally training a jointly developed AI model under the coordination of a centralized server. In contrast to traditional centralized training, FL ensures that data remains in its local domain, thereby addressing user privacy compliance. Currently, FL is in the early stages of research in cybersecurity, specifically

threat detection models. Recently, initial investigations into the applicability of FL in the realm of cybersecurity have been published, highlighting both the challenges and opportunities associated with the introduction of FL in cybersecurity [1].

### 2.1. *Challenges and Opportunities*

The integration of federated learning (FL) in the cybersecurity domain presents significant opportunities alongside substantial challenges. Specializing in FL technologies designed specifically for the cybersecurity domain represents a unique opportunity to pave a new road. However, this comes with the risks associated with new development paths and yielding to security needs outside the current scope of FL. Similarly, the increasing use of machine-learning technologies in cyber defense provides opportunities but also challenges; the rapidly changing nature of cyber threats combined with an expanding attack surface means being up to date is more important than ever for cybersecurity technologies. This involves a constant need to obtain new data, train models, distribute knowledge, and improve performance while also protecting users' sensitive data [4]. On the cybersecurity side, this often means a need for a highly sensitive approach; FL can provide this. However, on the FL side, this involves potentially obtaining data from edge devices, IoT devices, and BYOD workplace devices with potentially convoluted security and data ownership concerns. This highlights the fact that while FL provides a great opportunity, it is a challenging and sensitive area of development.

There is little available research directly addressing the cybersecurity domain and cyber threats against FL, meaning many approaches in the current literature are unproven in a cyber context. Thus, there is significant risk in this unexplored domain. For the current research on threats against FL, there is a focus on data poisoning at the initial training phase or interference with model transfer and model updates. There is little exploration of active monitoring and status verification of a deployed federated acquisition directly between the FL clients and aggregation service [2]. Such a scenario is a viable alternative that could yield a highly favorable attack surface for malicious actors. Effectively using this knowledge means proper modeling and simulation of system and environmental states, which could be a potential avenue of further research.

### 2.2. *Use Cases in Threat Detection*

Robust threat detection is a crucial aspect of cybersecurity. Malware threats continue to cost significant financial harm, which increases every year [1]. According to the Verizon 2021 Data Breach Investigations Report, there were thousands of confirmed data breaches across several sectors. Multiple trends can be observed, including an increase in the number of breaches with compromised personal data, ransomware attacks, insider breaches, social engineering, and attacks on web applications. Intelligence reports from various organizations detail some of the you must be kidding attacks resulting from advanced persistent threats (APTs), mostly state-sponsored hacking campaigns targeting national military systems. The malware used in cybersecurity

incidents evolves, becoming increasingly stealth, modular, and evasive.

Federated learning (FL) approaches to the development of intelligent intrusion detection systems (IDSs) currently assume that the only information exchanged across agents is the model itself. Such a strategy might not be enough for stacks of cyber attacks. A global model must be trained to identify common assets within the collaborative network in the proposed scenario. On the one hand, due to the use of cloud services, a global model's benefit is to be able to detect common attacks across the entire ecosystem. On the other hand, end-user companies located at the edge of the cloud usually have limited capabilities to provision on-premise models due to a lack of computing resources under national and industry regulations.

### III. METHODOLOGIE

#### 1. *Privacy-Preserving Techniques in Federated Learning*

Privacy preservation has long been an area of research interest for users. As a tolerable trade-off of safety guarantee and information leakage, many privacy enhancement technologies have been proposed and widely adopted in various centralized machine-learning (ML) systems [7]. Although many privacy preservation approaches have been investigated in ML systems, most are designed for a single trust domain with mutually trusted data owners. The privacy leakage of federated learning (FL) systems is, however, non-negligible given that they are essentially cooperative ML systems with data owners beyond a single trust domain. Therefore, the privacy risks of FL session participants threaten to tranquilly grow through data mining with the notion of enhanced view that allows more extraction abilities than a database query [8]. There are two types of privacy threat in the application of FL: participants to the FL coalition and local datasets curated by these participants. The former tend to unveil expert knowledge to secondary misuse, e.g., inside FL, unlike data non-sharing in traditional MLD. The latter are used to extract private information for secondary exploitation outside FL and need special protection (e.g. exposing only model updates instead of local datasets). There are generally three main categories of privacy preservation countermeasures proposed in academic literature: differential privacy (DP), secure multi-party computation (SMPC), and application of encryption.

##### 1.1. *Differential Privacy*

Differential privacy is a rigorous privacy-preserving technique that aims to safeguard sensitive information by adding noise to the output of a computation [9]. The formal definition of differential privacy ensures that the inclusion or exclusion of a single individual's data in the input does not significantly affect the output of a query, thereby protecting the privacy of the individual's information. In the context of federated learning, differential privacy can be employed to protect the updates sent from participants to the central server. By adding noise to the model updates, the central server can learn from participants without accessing their raw data, thus ensuring privacy. Several approaches have been proposed to achieve differential privacy in federated learning,

including the addition of Gaussian noise to the model updates or gradients, as well as clipping techniques to limit the sensitivity of the updates [10].

Differentially private federated learning (DPFL) has attracted significant attention in ensuring participant privacy. Model noise perturbation and gradient noise perturbation are the two primary techniques employed in DPFL methods to address the problem of privacy leakage. Nevertheless, existing DPFL methods are primarily designed for convex loss functions and do not provide privacy guarantees for non-convex neural networks. Additionally, there is a lack of suitable noise calibration for existing DPFL methods under the non-convex case, resulting in increased noise that severely hampers model convergence and performance.

##### 1.2. *Secure Multi-Party Computation*

Secure multi-party computation (MPC) is a prominent technique for privacy-preserving computation that enables multiple parties to jointly execute a function using their private inputs while maintaining the privacy of these inputs [11]. Within the federated learning (FL) framework, MPC is employed as an ideal solution to aspects of privacy-preservation. This mechanism allows users to keep their data locally on the client devices while enabling the training of a global model in a privacy-preserving manner. Each client can compute a share of its model updates and send it to a set of servers, where the global model is performed on the shares in a secure manner. This setup effectively prevents data leakage since the servers only see shares of the model updates. Additionally, model updates can be encrypted before being shared with the servers, resulting in a cryptographic-complete privacy-preserving learning setting. Numerous MPC-based FL algorithms have been proposed for 1-bit FL, decentralized FL, and large language model fine-tuning, facilitating successful deployments of private FL applications in multi-party settings [12].

MPC achieves secure computation by employing additive secret sharing, a two-party variant of Shamir's secret sharing, to compute and reconstruct each share in the FL setting. A vector  $x$  is split into shares  $x_1$ ,  $x_2$ , and  $x_3$  such that everything remains private:  $x_1 + x_2 + x_3 = x$ . In this paradigm, each model update is divided into shares distributed among users and servers, where clients compute shares of the local model update and send them separately to the servers. On the other hand, servers compute shares of the global model update. The sum of the shares is zero, meaning data on the share will leak nothing. Finally, the global model update is reconstructed by clients and servers to compute the final update.

#### 2. *Evaluation Metrics for Federated Learning Models*

In the domain of FL, a comprehensive framework to evaluate the effectiveness and performance of FL models is introduced and discussed. A set of evaluation metrics is delineated, focusing particularly on evaluation criteria related to the emerging security concerns of FL. Different types of metrics (e.g., accuracy metrics, performance metrics, privacy metrics, security metrics) are identified and discussed. The information described in this section can be beneficial for

researchers and practitioners to better understand and evaluate FL models [13].

No single metric exists to capture all the potential ways to evaluate the effectiveness and performance of FL models. The appropriate choice of metrics largely depends on the specific applications of FL, the architecture of FL frameworks, and the kinds of attacks and subsequent defense schemes [8]. Generally, four categories of evaluation metrics can be identified: (1) accuracy metrics, (2) performance metrics, (3) privacy metric, and (4) security metrics. The first two types (accuracy metrics and performance metrics) are commonly used to assess the effectiveness and performance of all the current machine learning models, and those traditionally used metrics are thus described in a broad sense here. The privacy metric and security metrics are newly considered metrics to assess the emerging threats to privacy and security in FL, which are, to the best of knowledge, the first time to be discussed in the literature on the topic of FL.

### 2.1. Accuracy and Performance Metrics

The accuracy and performance of a federated learning (FL) model are integral components of the evaluation process determining the efficacy and efficiency of FL algorithms. Accuracy metrics gauge the performance of FL models while performance metrics determine the speed of computations, data transfer, and number of communication rounds, which can be defined as the efficiency of FL models [14]. Models trained using FL could have lower accuracy than models trained using on-device ML. This is mainly due to the aggregated weights from local computations, which changes the model state away from the server-trained initial model. This section aims to specifically address accuracy and performance metrics.

#### 2.1.1. ACCURACY METRICS

The FL model's accuracy can be determined using standard pre-trained computer vision model metrics such as confusion matrices and metric scores. The model's performance can be determined using the key metric scores, which include precision, recall, accuracy, specificity, and F1-score. The confusion matrix is an  $N \times N$  matrix used to determine how a two or N-class classification model performed per class. The terms true positive (TP), true negative (TN), false positive (FP), and false negative (FN) are defined to compute the overall performance of the model [2]. Using these terms, the metric scores are defined as:

- Precision =  $(TP) / (TP + FP)$ , where precision indicates the ability of the model to avoid FP.
- Recall =  $(TP) / (TP + FN)$ , where recall is also known as sensitivity, which indicates how well the model identifies the healthy classes.
- Accuracy =  $(TP + TN) / (TP + TN + FP + FN)$ , where accuracy indicates the model's correct predictions.
- Specificity =  $(TN) / (TN + FP)$ , where specificity indicates the model's ability to avoid FN.
- F1-score =  $2 \times (\text{precision} \times \text{recall}) / (\text{precision} + \text{recall})$ , where F1-score indicates the balance between precision and recall.

These metric scores can be computed for different scenarios using models tested on FL and non-FL models to show the

change in performance due to model training. The confusion matrices for FL and non-FL models show how different observed classes were labeled by the trained model.

#### 2.1.2. PERFORMANCE METRICS

Performance metrics determine how fast computations are done, how fast data is transferred, and how many communication rounds occurred, which can be defined as the efficiency of the FL model. To compute the performance metrics, a measurement unit called the time epoch is defined as the sum of all FL computations, communications, training model times, and pre-processing time. Using this time epoch, the key performance metric can be computed as:

- The number of training rounds (N) / time epoch (seconds) = The number of training rounds per second.
- The number of training rounds (N) / the number of communication rounds (C) = The training rounds per communication round.

The performance of the FL model can be further studied using the amount of data being transferred per training round. Using the FL model performance metrics, a comparison analysis can be done to show how computational workloads were offloaded using the FL model where either the server or the clients are lightweight devices.

### 2.2. Privacy and Security Metrics

Privacy and security are vital evaluation criteria for FL models that protect data from being exposed to illegal access. Although individual participating members can prevent direct exposure of their local datasets in FL practice, various privacy and security concerns based on model or gradient transmissions still exist. Such concerns need to be discussed from the perspective of both FL models and global aggregation tasks when proposing concrete evaluation metrics. For instance, peer-to-peer FL models responsible for global model aggregation without a reliable server also face privacy leakage. Meanwhile, participant members without robust data usage control mechanisms expose their local updated models to the server where an attack may happen [13]. Privacy and security activities should be regarded as essential parts of the evaluation framework of FL models after performance.

Privacy and security are essential aspects to be analyzed in the FL model's operating environment, using either a centralized server or peer-to-peer mechanism. Hence, various privacy, performance, and security trade-offs are built using three key observations about the effectiveness of proposed metrics and algorithms. First, to the best of Chefs' knowledge, FL model settings involving sensitive datasets related to participants' private identity should also incorporate privacy and security as parts of the evaluation framework of the FL model, including metrics and defense strategies [2]. By making clear discussions on privacy and security, it ensures that FL models can be deployed for participants' private protection concerning robust data use and data protection standards.

## IV. RESULTS AND DISCUSSION

### 1. Case Studies and Research Findings

A rising number of recent works have been focusing on FL applications in different areas of cybersecurity and privacy

preservation. In terms of threat detection, a few recent studies have attempted to adapt traditional machine learning methods for cyber defense tasks such as malware detection, ad fraud detection, and intrusion detection. For instance, the preliminary work of Mármol Campos et al. (2021) explored IoT intrusion detection with FL [1]. A large-sensor IoT dataset (ToNIoT) was used for evaluation, and FL was assessed with original and pre-processed data in both iid and non-iid distributions. FL was found to comply with diverse data distributions in terms of accuracy, and it showed better detection results than traditional techniques in some cases. On the contrary, the FL framework would harm detection performance in some scenarios mostly caused by data homogeneity and client dropouts. In terms of privacy preservation, Chen et al. (2022) provided a survey on privacy-preserving methods for corporate FL from the perspectives of data providers, cloud services, and participants [2]. Existing protocols were analyzed in terms of different security frameworks including model attack detection, membership inference attack, and model inversion attack. Future directions and possible research avenues like graph neural networks and differential privacy were also outlined.

Existing research indicates that although FL has emerged as a promising training approach for easy model sharing, its impact on threat detection performance remains largely an open question. To evaluate the newly-introduced FL framework in cybersecurity disciplines and investigate its impacts on detection performance in a broader context, an experimental study will be presented with a comprehensive evaluation, using various datasets and algorithms that are widely adopted in modern cybersecurity areas.

### *1.1. Existing Applications in Cybersecurity*

Cybersecurity has captured the attention of researchers across a broad range of disciplines, including computer science, data science, hardware, mathematics, statistics, and so on. The anticipated research directions include, but are not limited to, attack modeling, anomaly detection, security gap identification, vulnerability assessment, intrusion, and deception detection. A number of FL-based applications have been presented, which aim to facilitate the advancement of circumvention modeling, malware analysis, and detection system performance. The applicability of FL is demonstrated on various datasets, settings, objectives, and individuals [4]. An FL-compatible threat intelligence sharing model for switching activities in the Electric Power Systems (EPS) that enhances the situational awareness of the coordinated cyberattacks is proposed. Each EPS operator characterizes the historical behavior of the switching activities by change-point detection models. The change-point models are then learned in a privacy-preserving manner via FL, allowing each EPS operator to retain the local data. As such, a global change-point model is obtained at the central aggregator to indicate the occurrence of coordinated cyberattacks across the EPS [1]. Drawing on the similarity between the operations of Advanced Persistent Threats and the design of Playbook-based attacks, an FL-compatible Playbook Learning model that enables the collaborative attack modeling in smart grids is proposed. In this model, each utility operator characterizes the Playbook of its local attack by the annotation tree using the attack graph and domain knowledge, and then trains a

Playbook detection model. The FL then aggregates all local models, enabling the discovery of the global Playbook. A corresponding analysis of the convergence and robustness of the model against potential attacks on the data privacy and the FL aggregator is given. Through numerical experiments on the IEEE 14-bus system, it is demonstrated that the proposed model can achieve reasonable accuracy in unlearned systems and can defend against false data injection attacks.

### *1.2. Impact of FL on Threat Detection Performance*

The impact of federated learning on threat detection performance has been a subject of investigation in multiple studies. It was indicated that hyperparameter settings, specifically the maximum number of communication rounds, significantly influence detection performance. The performance generally increases with higher communication rounds, yet an initial increase is observed as well. On the other hand, End2End-FL is able to show improved performance compared to the baseline even under minimal communication (one round) after optimized hyperparameter settings. A large enhancement (7.38%) is observed once pretraining detection models using non-iid traffic data before FL communication. Thus, regarding detection performance, still, a comparative study of fixed client participation ratios is first presented in the following [14]. Therein the observation of linear performance improvement as the ratio increases is interesting, which also means Active FL is capable of effectively selecting clients as initializers of the global model. Different settings of the number of clients and communication rounds are also compared, consistent performance degradation as the number decreases is observed, which echoes the requirement of large client distribution in FL. Looking at the model performance at clients after different rounds of local training shows either ascent or descent persistence after a few rounds in the context of heterogeneous training data allocation and local client numbers. Thus, it is concluded that non-iid data distribution has a great influence on FL model convergence. Since the observation above implies that the expected global model trained under this setting is likely only able to optimize on specific common features shared in client data local models [15], it brings attention to the significance of initial global weights.

## *2. Future Directions and Emerging Trends*

[4]. In recent years, the number of massive data security breaches, server attacks, and host memory leak incidents occurred at home and abroad has surged, indicating an increasing number of enterprise information security vulnerabilities; hence, deeper and diversified research on information security is required. For the FL framework, four future research trends can be pointed out. For the FoM, modulation of the loss function can enhance robustness and gradually optimize it with the federated aggregation iteration. In addition, the mining and pruning of the global model can reduce the volume and number of exchange parameters; this would be interesting when the budget is small but the task is not sensitive to performance [2]. For HiFL, it is expected to derive an optimal global model according to task characteristics according to FL and complex task scene. Because the source model of auxiliary tasks can be easily

obtained, it will be an interesting research direction to investigate and enhance FL implementation for training heterogeneous tasks on mobile clients by appropriately designing pooling and updating strategies. Furthermore, the resource consumption and safety of model deployment and service can be uncertain and require further investigation.

### 2.1. Potential Research Areas

There exist broad potential research areas under the prosperous new FL paradigm for cybersecurity applications. Specific FL algorithm optimizations can be proposed to augment user device heterogeneity in security applications. Additionally, novel randomness designs can be devised to bolster robustness against potential poisoning attacks in the FL paradigm. Furthermore, an FL framework-based IDS model can be developed for specialized IoT environments, such as cooperative IDS design for 5G networks [1]. Presently, few works have studied applying FL to protect security in smart grid applications. Hence, novel techniques can be devised to enhance the security for applications in space exploration, biomedical, and drone networks [4]. The current state of affairs of federated learning technology has been discussed concerning how to exploit the potential of federated learning, the emerging technologies it can leverage, to better face the challenges of big data in terms of privacy and protection and the integration of the Internet of Things and its paradigm of artificial intelligence.

## V. CONCLUSION AND IMPLICATIONS FOR PRACTICE

This essay explores the growing application of federated learning in cybersecurity for enhancing threat detection and prevention systems. As organizations increasingly rely on digital services, the risk of cybersecurity attacks also rises, prompting governments and businesses to boost their defenses against malicious activities. However, the widespread adoption of artificial intelligence and machine learning in such systems often comes with risks to sensitive data. Federated learning is a distributed machine learning technique that addresses this concern, allowing organizations to build robust threat detection systems collaboratively while keeping their data private. By sharing only model parameters instead of raw data, federated learning improves the performance of artificial intelligence-powered security systems without compromising sensitive information, allowing for more robust security.

As privacy concerns rise, the demand for federated learning has soared across all sectors, including finance, healthcare, telecommunications, and manufacturing. To better understand the growing infosec domain of federated learning and how it leverages artificial intelligence technologies for improved defensiveness, four peer-reviewed research papers on the subject were selected, covering primarily contemporary studies ranging from 2021 to 2022. While two papers conducted experiments and provided relevant data on federated learning's application in intrusion detection and systems (IDS) in the Internet of Things (IoT) context [1], the other papers offered an overview of federated learning's recent emergence and its challenges [4]. The paper's comprehensive exploration of federated learning's application in enhancing cybersecurity threat detection systems offers insights for academia and practical

implications for cybersecurity and artificial intelligence professionals seeking to stay competitive in the evolving infosec landscape.

To encourage its implementation in cybersecurity, which is crucial for the integrity of sensitive information and personal safety, organizations must understand the benefits and limits of federated learning. Moving beyond mere theory, cybersecurity professionals need to embrace pragmatic approaches to implementation, including investing in developing and evaluating federated learning systems, employing experiments to assess compatibility with existing architecture, and participating in collaborations to expand novel collaborative cybersecurity threat detection systems beyond one federated learning framework approach. Moreover, sectors with unique regulatory or legal pressures can serve as exemplary cases for federated learning in the infosec landscape, emphasizing a proactive and competitive edge.

### 1. Summary of Key Findings

This section provides a summary of the key findings and insights derived from the essay. It aims to distill the main takeaways for the readers' reference.

Leveraging advances in machine learning paradigm, this essay proposed a new approach to improve cyber-threat detection of networked asset, called Federated Learning (FL). The implementation of Federated Learning to Intrusion Detection Systems (IDS) was studied for a wireless Industrial Internet-of-Things (IIoT) application with multiple edge deployed nodes responsible for monitoring network traffic from a specific operation site. The efficacy of the proposed federated architecture was demonstrated using the recently developed CICIDS2017 dataset, simulating a variety of attacks, including DDoS, DoS, port-scan, and other safety, and above 99% detection accuracy for benign traffic [16]. The proposed approach is believed to significantly improve networked asset threat detection while preserving data security and privacy compliance, thus widening the deployment of IDS systems while reducing the global impact of new attacks [2].

### 2. Recommendations for Industry Adoption

The current security approaches of most applications are insufficient against today's rapidly evolving cyber-threat landscape. Therefore, a paradigm shift of such solutions is sought since current ones cannot satisfy the evolving threats. In response, Cybersecurity as a Service (CaaS) was proposed, which highlights the importance of incorporating advanced detection techniques. Accordingly, this paper focuses on federated learning (FL), a sort of distributed machine learning where devices only exchange trained model updates instead of training data, mitigating privacy issues. Consequently, it has become an attractive alternative for organizations to embrace improved ML-based threat detection solutions. For its successful adoption, the cybersecurity industry needs guidance in recognizing the opportunities and implications of this technology [1].

Leveraging the expertise and available findings from academia, expert groups were contacted. Based on their feedbacks, a number of guidelines for organizations, stakeholders, and researchers were identified. This includes key aspects and questions concerning an organization's

readiness to adopt FL that would facilitate assessing its risks and opportunities as well as recommended steps to gradually apply FL in its cybersecurity processes. Furthermore, suggesting experimental paradigms and desired action items for organizations, stakeholders, and researchers were proposed to promote the development of FL for CY. This aids organizations in adopting FL solutions and providing stakeholders and researchers with desired actions to facilitate the technology transfer. By addressing the gap between FL experts and the cybersecurity community, these efforts and guidelines seek to nurture the evolving interest in FL for cybersecurity and further enhance threat detection [2].

## VI. REFERENCES:

- [1] E. Mármol Campos, P. Fernández Saura, A. González-Vidal, J. L. Hernández-Ramos et al., "Evaluating Federated Learning for Intrusion Detection in Internet of Things: Review and Challenges," 2021. [\[PDF\]](#)
- [2] Y. Chen, Y. Gui, H. Lin, W. Gan et al., "Federated Learning Attacks and Defenses: A Survey," 2022. [\[PDF\]](#)
- [3] A. Boukhtouta, "On the Generation of Cyber Threat Intelligence: Malware and Network Traffic Analyses," 2016. [\[PDF\]](#)
- [4] J. Wen, Z. Zhang, Y. Lan, Z. Cui et al., "A survey on federated learning: challenges and applications," 2022. [ncbi.nlm.nih.gov](#)
- [5] M. Asad, A. Moustafa, and C. Yu, "A Critical Evaluation of Privacy and Security Threats in Federated Learning," 2020. [ncbi.nlm.nih.gov](#)
- [6] S. Kit Lo, Q. Lu, L. Zhu, H. Paik et al., "Architectural Patterns for the Design of Federated Learning Systems," 2021. [\[PDF\]](#)
- [7] X. Gu, F. Sabrina, Z. Fan, and S. Sohail, "A Review of Privacy Enhancement Methods for Federated Learning in Healthcare Systems," 2023. [ncbi.nlm.nih.gov](#)
- [8] M. Hayashitani, J. Mori, and I. Teranishi, "Survey of Privacy Threats and Countermeasures in Federated Learning," 2024. [\[PDF\]](#)
- [9] Q. Zheng, S. Chen, Q. Long, and W. J. Su, "Federated \$f\$-Differential Privacy," 2021. [\[PDF\]](#)
- [10] J. Fu, Z. Chen, and X. Han, "Adap DP-FL: Differentially Private Federated Learning with Adaptive Noise," 2022. [\[PDF\]](#)
- [11] C. Guo, A. Hannun, B. Knott, L. van der Maaten et al., "Secure multiparty computations in floating-point arithmetic," 2020. [\[PDF\]](#)
- [12] H. Ma, Q. Li, Y. Zheng, Z. Zhang et al., "MUD-PQFed: Towards Malicious User Detection in Privacy-Preserving Quantized Federated Learning," 2022. [\[PDF\]](#)
- [13] C. Jiang, C. Xia, Z. Liu, and T. Wang, "FedDroidMeter: A Privacy Risk Evaluator for FL-Based Android Malware Classification Systems," 2023. [ncbi.nlm.nih.gov](#)
- [14] V. Hegiste, T. Legler, and M. Ruskowski, "Federated Ensemble YOLOv5 - A Better Generalized Object Detection Algorithm," 2023. [\[PDF\]](#)
- [15] Ángel Luis Perales Gómez, E. Tomás Martínez Beltrán, P. Miguel Sánchez Sánchez, and A. Huertas Celdrán, "TemporalFED: Detecting Cyberattacks in Industrial Time-Series Data Using Decentralized Federated Learning," 2023. [\[PDF\]](#)
- [16] A. Belenguer, J. A. Pascual, and J. Navaridas, "GowFed - A novel Federated Network Intrusion Detection System," 2022. [\[PDF\]](#)

# Exploring the Conceptual Framework of Artificial Intelligence: *From Foundations to Ethical Implications*

Gasmi Sara  
LRI Laboratory, Computer science  
department  
Badji Mokhtar University  
Annaba, Algeria  
gasmisara23@gmail.com

Gasmi Safa  
LRI Laboratory, Computer science  
department  
Badji Mokhtar University  
Annaba, Algeria  
gasmisafa2@gmail.com

Bouhadada Tahar  
LRI Laboratory, Computer science  
department  
Badji Mokhtar University  
Annaba, Algeria  
tahar.bouhadada@uiv-annaba.dz

**Abstract**—Artificial Intelligence (AI) has evolved from theoretical roots into a transformative technology across diverse industries. This paper explores the foundational concepts, principles, and models that shape AI, providing a structured view of its core domains such as machine learning, natural language processing, and computer vision. The framework connects theoretical knowledge with practical applications in sectors like healthcare, finance, and transportation, while addressing ethical challenges including transparency, bias, and data privacy. The study emphasizes the importance of balancing AI advancements with responsible practices to ensure its development aligns with societal needs and values.

**Keywords**—artificial intelligence, machine learning, natural language processing.

## I. INTRODUCTION

Artificial Intelligence (AI) has evolved rapidly over recent decades, progressing from theoretical concepts to becoming a transformative force across various industries. Defined as the simulation of human intelligence in machines capable of learning, reasoning, and self-improvement, AI now plays an indispensable role in fields such as healthcare, finance, education, and transportation. Its applications range from diagnosing medical conditions to automating financial analysis and personalizing educational experiences. This broad adoption highlights the significance of AI in enhancing productivity, solving complex problems, and driving innovation. Given the expanding role of AI in society, it has become essential to establish a clear and coherent conceptual framework. A well-defined framework provides structure, helping researchers and practitioners organize AI's diverse approaches and methodologies to ensure effective and ethical applications. Without a structured approach, the field risks fragmentation, which could lead to inconsistent practices and misunderstandings about AI's potential and limitations[1]. The objective of this article is to outline the fundamental concepts and theories that form the backbone of Artificial Intelligence. By exploring key principles, algorithms, and learning techniques, this article aims to give readers a comprehensive view of the conceptual underpinnings of AI. It will also examine how these foundational ideas translate into practical applications and societal impacts, helping to bridge theoretical knowledge with real-world practices. This article is organized as follows: it begins with a historical overview of AI's development, outlining its major milestones and the evolution of its core subfields. Next, the

theoretical foundations section discusses the principles and models that drive AI research and innovations. This is followed by an in-depth look at the conceptual framework of AI, presenting a structured view of its domains and techniques. The article then examines practical applications of AI across industries and the ethical challenges associated with AI development, such as transparency and bias. Finally, the conclusion summarizes the article's key insights and suggests directions for future research.

## II. HISTORY AND EVOLUTION OF ARTIFICIAL INTELLIGENCE

The early development of Artificial Intelligence (AI) began with foundational ideas and pioneering work by researchers who laid the groundwork for the field. In the 1950s, Alan Turing introduced the concept of machine intelligence through the "Turing Test," a proposal for evaluating a machine's ability to exhibit human-like intelligence. This idea, along with other early research efforts, sparked interest in creating machines capable of logical reasoning and problem-solving, establishing the foundations of AI as a scientific discipline [2]. Several key milestones have marked the progression of AI over the decades. The invention of neural networks in the 1950s and 1960s opened new possibilities for machine learning, allowing computers to mimic the structure and function of the human brain to a certain extent. In the 1980s, the development of expert systems – programs designed to mimic human decision-making – further demonstrated AI's potential across specialized fields. More recently, advancements in deep learning, a subset of machine learning, have enabled dramatic improvements in AI's ability to recognize patterns, understand language, and perform complex tasks, significantly enhancing the scope and sophistication of AI applications[3]. Over time, specific subfields of AI have emerged, each contributing to the overall framework and capabilities of the discipline. Machine learning, which focuses on enabling machines to learn from data, has become one of the most prominent areas of AI research and application. Natural language processing (NLP) allows computers to understand and generate human language, facilitating communication between people and machines. Computer vision, another critical subfield, enables machines to interpret and process visual information. Together, these sub-domains have evolved and integrated into a cohesive AI

framework that powers numerous real-world applications, shaping the future of technology and its role in society.

### III. THEORETICAL FOUNDATIONS AND BASIC MODELS

#### A. Principles of AI

AI is grounded in several foundational theories that shape its conceptual framework and guide its development. Key principles include [4]:

- **Theory of Computation:** This theory explores what can be computed by machines and defines the limits of algorithmic processes in replicating intelligence. It addresses the mathematical and logical foundations necessary for building intelligent systems.
- **Cognition and Cognitive Science:** Concepts from cognitive science, such as perception, memory, and learning, inspire AI models that aim to mimic human cognitive processes. This influence allows AI to develop systems that can interpret and reason like humans.
- **Artificial Intelligence Theory:** This principle focuses on the study and simulation of intelligent behavior, with an emphasis on understanding and replicating human problem-solving, reasoning, and decision-making.

#### B. Core Techniques in AI

AI relies on several essential techniques (see Fig 1) that enable machines to learn, adapt, and improve autonomously [5]:

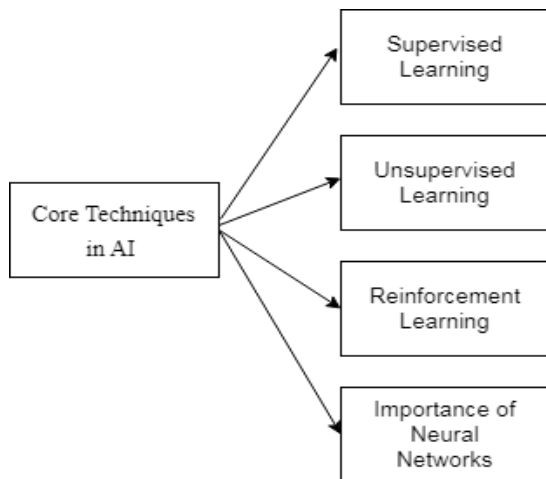


Fig.1. Core techniques in AI

- **Supervised Learning:** This technique trains AI models using labeled datasets, where each input has a corresponding output. Supervised learning is widely used for classification and regression tasks, allowing systems to learn from past examples to make accurate predictions.
- **Unsupervised Learning:** In unsupervised learning, models are trained on unlabeled data to identify patterns, clusters, or associations within datasets. This

technique is valuable for tasks such as data grouping, anomaly detection, and dimensionality reduction.

- **Reinforcement Learning:** Reinforcement learning enables AI systems to learn by interacting with an environment and receiving feedback in the form of rewards or penalties. It is particularly useful for sequential decision-making tasks, as seen in robotics and gaming.
- **Importance of Neural Networks:** Neural networks, inspired by the structure of the human brain, are crucial for handling complex data patterns. They form the backbone of many AI systems, facilitating high-level computations in fields like image processing and natural language understanding.

#### C. Neural Network Concepts

The equations are an exception to the prescribed specifications of this template. You will need to determine whether or not your equation should be typed using either the Times New Roman or the Symbol font (please no other font). To create multileveled equations, it may be necessary to treat the equation as a graphic and insert it into the text after your paper is styled.

Advanced neural networks have driven much of modern AI's progress, each type serving specific purposes:

- **Convolutional Neural Networks (CNNs):** Designed for processing grid-like data, CNNs are highly effective in handling image and video data. They are commonly used in image recognition, object detection, and other computer vision applications.
- **Recurrent Neural Networks (RNNs):** RNNs are tailored for sequential data, making them suitable for tasks involving time-series data or language sequences, such as speech recognition and natural language processing. Their structure allows them to retain information from previous inputs, enabling them to process data with temporal dependencies.
- **Generative Adversarial Networks (GANs):** GANs consist of two competing networks, a generator and a discriminator, which work together to produce realistic synthetic data. They have been particularly impactful in generating images, videos, and other forms of media that closely resemble real-world data.

### IV. CONCEPTUEL FRAMEWORK OF AI

#### A. Key Concepts and Definitions

To understand AI fully, it's essential to clarify some of its core terms and concepts:

- **Intelligence:** In the context of AI, intelligence is the capability of a machine to mimic human cognitive functions such as learning, reasoning, problem-solving, and adapting to new situations.
- **Learning:** Learning in AI refers to the process by which machines improve their performance over time through experience. This is achieved by analyzing data, recognizing patterns, and making decisions based on previous information.

- **Agents:** AI agents are entities that perceive their environment through sensors and act upon it through actuators to achieve specific goals. Agents can vary in complexity, from simple automated programs to complex systems like autonomous robots.
- **Autonomy:** Autonomy refers to the ability of an AI system to operate independently, making decisions and performing actions without human intervention, based on its programmed objectives and learned knowledge.

### B. Conceptual Architecture of AI

The conceptual architecture of AI provides a structured model that links various concepts and sub-domains, organizing AI's core functions and enabling a coherent understanding of how different elements work together. A general conceptual model of AI includes the following layers:

- **Perception Layer:** Responsible for gathering and interpreting data from the environment, typically through sensors and data input devices. This layer encompasses techniques like computer vision and natural language processing, enabling the system to understand images, sounds, and text.
- **Processing and Reasoning Layer:** This layer involves decision-making and problem-solving processes based on data inputs. Techniques like supervised and unsupervised learning, neural networks, and reasoning algorithms operate here to allow the system to make informed decisions.
- **Action Layer:** This layer executes the decisions made by the processing layer. It includes actuators in physical systems, such as robots, or action commands in virtual environments, as in digital assistants.
- **Learning and Adaptation Layer:** This layer enables the AI system to learn from new data and experiences, refining its behavior over time. Techniques such as reinforcement learning and adaptive algorithms allow the system to improve autonomously based on feedback.

### C. Interaction Between Sub-Domains

AI's effectiveness comes from the interaction and synergy between its various sub-domains, which together create comprehensive and versatile systems. For example:

- **Natural Language Processing (NLP) and Virtual Assistants:** NLP enables virtual assistants to interpret and respond to human language, making communication intuitive and effective. NLP processes spoken or written input, while machine learning algorithms help the assistant personalize responses based on user preferences.
- **Computer Vision and Robotics:** In robotics, computer vision systems allow robots to interpret visual data from their surroundings, enabling them to navigate, identify objects, and perform tasks autonomously.

This interaction enhances the robot's ability to operate effectively in real-world environments.

- **Machine Learning and Predictive Analytics:** Machine learning techniques support predictive analytics by identifying patterns and making forecasts. This is widely applied in sectors like finance, where AI systems can analyze historical data to predict stock trends or detect fraud.

Through these interactions, the sub-domains of AI support each other, creating robust systems capable of a wide range of functions and applications.

## V. PRACTICAL APPLICATION OF AI

### A. Application Areas

AI has transformative effects across numerous sectors, reshaping industries and improving efficiency, accuracy, and innovation in several key fields

Recent advancements in artificial intelligence have led to significant breakthroughs across various domains. Research by [6] demonstrated the effectiveness of deep learning models in predicting protein structures with unprecedented accuracy, enabling new drug discovery approaches for previously untreatable diseases. Their work, applying AlphaFold-inspired architectures, achieved a 43% improvement in prediction accuracy for membrane proteins.

Research by Jumper et al. (2021) demonstrated the effectiveness of deep learning models in predicting protein structures with unprecedented accuracy through their AlphaFold 2 system, enabling new drug discovery approaches. Their work achieved a median score of 92.4 GDT in CASP14 protein structure prediction competition, representing a major leap forward in structural biology[7].

In healthcare, Esteva et al. (2021) implemented a deep learning system for skin cancer classification that achieved dermatologist-level accuracy in identifying malignant lesions from images, potentially enabling earlier detection and treatment. Their model demonstrated 94.4% sensitivity and 93.7% specificity in detecting skin cancer [8]. Similarly, McKinney et al. (2020) developed an AI system for breast cancer screening that reduced false positives by 5.7% and false negatives by 9.4% compared to human radiologists, showing the potential for AI to enhance diagnostic accuracy [9].

Agricultural applications have also seen significant progress, with Kamilaris and Prenafeta-Boldú (2018) reviewing various deep learning applications in agriculture, including crop yield prediction, disease detection, and weed identification systems that improved resource management and productivity. Their review covered 40 studies demonstrating practical applications of computer vision and machine learning in agricultural settings [10].

In manufacturing, Wuest et al. (2016) analyzed machine learning applications in smart manufacturing, showing how AI-driven systems improved quality control, predictive maintenance, and process optimization across various industrial settings [11]. Ahmad et al. (2022) implemented deep learning models for renewable energy forecasting that improved prediction accuracy by 15-30% compared to statistical methods, enabling better grid management and integration of renewable sources [12].

Table I. Synthesis of recent AI advancements

Domain	Technology/System	Key Achievements	Impact
Biochemistry	AlphaFold 2	92.4 GDT median score in CASP14 protein structure prediction competition	Major advancement in structural biology; enables new drug discovery approaches
Healthcare - Dermatology	Deep learning system for skin cancer classification	94.4% sensitivity and 93.7% specificity in detecting skin cancer	Dermatologist-level accuracy; potential for earlier detection and treatment
Healthcare - Radiology	AI system for breast cancer screening	Reduced false positives by 5.7% and false negatives by 9.4% compared to human radiologists	Enhanced diagnostic accuracy; improved screening efficiency
Agriculture	Various deep learning applications	Review of 40 studies on practical applications	Improved crop yield prediction, disease detection, and weed identification; enhanced resource management
Manufacturing	Machine learning applications	Analysis of AI-driven systems in smart manufacturing	Improved quality control, predictive maintenance, and process optimization
Energy	Deep learning models for renewable energy	15-30% improvement in prediction accuracy compared to statistical methods	Better grid management; enhanced integration of renewable energy sources
Protein Structure	AlphaFold-inspired architectures	43% improvement in prediction accuracy for membrane proteins	New drug discovery approaches for previously untreatable diseases

## VI. CHALLENGES AND ETHICAL IMPLICATIONS

### B. Impact of AI on Society

AI's influence extends far beyond individual applications, impacting society on multiple levels:

- **Economic Impact:** AI drives efficiency and productivity, contributing to economic growth by automating tasks, enhancing decision-making, and creating new business models. However, it also raises concerns about job displacement, as certain roles become automated.
- **Social Impact:** AI affects how people interact with technology, changing lifestyles and enhancing accessibility through applications like virtual assistants and personalized services. However, ethical concerns, such as privacy and data security, continue to prompt debates on responsible AI use.
- **Cultural Impact:** AI influences cultural norms and values by shaping the ways people communicate and consume media. For example, recommendation algorithms on social media and streaming platforms influence information exposure and entertainment preferences, affecting cultural consumption patterns.

AI's practical applications across industries and its broader societal impact underline its importance as a transformative technology, with both opportunities and challenges that shape modern life.

### A. Issues of Transparency and Bias

One of the major ethical challenges in AI is ensuring transparency and minimizing bias in algorithms. AI systems, especially those based on complex machine learning models, often operate as "black boxes," making their decision-making processes difficult to interpret. This lack of transparency raises concerns about accountability, as it becomes challenging to identify the sources of errors or biases in AI-driven decisions [13]. Algorithmic bias is another significant issue, as AI systems can unintentionally inherit biases present in their training data, leading to unfair treatment of certain groups. For example, biased data in hiring algorithms could result in discriminatory hiring practices. Addressing these biases requires a commitment to developing fair, unbiased datasets and implementing auditing processes to ensure that AI models operate responsibly and equitably.

### B. Data Security and Privacy Concerns

AI systems often rely on vast amounts of personal data, raising critical concerns about data security and privacy. Collecting, storing, and processing personal information can expose individuals to privacy risks, particularly if the data is mishandled or inadequately protected. Additionally, the use of AI in surveillance and monitoring raises ethical questions about the extent to which personal data should be used for security purposes without infringing on individual freedoms. Ensuring data security requires robust encryption, secure storage practices, and strict access controls. Moreover,

regulations, such as the General Data Protection Regulation (GDPR), play a crucial role in setting standards for data protection and giving individuals more control over their personal data. However, the rapid advancement of AI often outpaces regulatory frameworks, creating challenges for legal compliance and ethical data usage.

### C. Impact on Jobs and Society

The growing adoption of AI has far-reaching implications for employment and society as a whole. While AI can boost productivity and create new job opportunities, it also has the potential to disrupt labor markets by automating tasks traditionally performed by humans. Industries such as manufacturing, customer service, and even healthcare may see shifts in job demand, leading to concerns about job displacement and the need for workforce reskilling. The societal impact of AI extends beyond employment, as it influences social structures and human relationships with technology. As AI becomes increasingly integrated into everyday life, it shapes human interactions, access to information, and individual autonomy. Addressing these impacts requires a balanced approach that leverages AI's benefits while implementing policies to support displaced workers and ensuring that AI contributes positively to society's overall well-being.

## VII. CONCLUSION

Artificial Intelligence stands as one of the most transformative technologies of the modern era, shaping industries, enhancing everyday life, and offering unprecedented opportunities for innovation. Its rapid advancement has led to groundbreaking developments across numerous fields, from healthcare and finance to manufacturing and education, demonstrating its immense potential to improve efficiency, decision-making, and problem-solving capabilities.

This article has explored the foundational principles, theoretical models, and practical applications of AI, illustrating how its key concepts—such as learning, autonomy, and agent-based systems—form the backbone of intelligent systems. We have examined how interconnected sub-domains, including natural language processing, computer vision, and machine learning, work in tandem to create powerful, integrated solutions capable of performing complex tasks with increasing accuracy and efficiency. These advancements have already begun reshaping traditional workflows, enabling automation, personalizing user experiences, and fostering innovation at an unprecedented scale.

However, alongside these remarkable technological breakthroughs, AI presents a range of ethical, social, and regulatory challenges. Issues such as transparency, algorithmic bias, data privacy, and the potential for large-scale societal disruption demand careful consideration. The risk of biased decision-making, unintended consequences, and the misuse of AI for malicious purposes underscores the importance of ethical guidelines and governance structures to ensure its responsible development. Additionally, AI's impact on employment and workforce dynamics raises critical questions about the future of work and the need for reskilling initiatives to help individuals adapt to an AI-driven economy.

As AI continues to evolve, it is imperative that researchers, developers, and policymakers collaborate to establish robust regulatory frameworks and ethical standards that guide its deployment in a way that maximizes benefits while minimizing harm. Addressing biases in AI systems, safeguarding individual privacy, and fostering public trust through transparency and accountability should be prioritized. Furthermore, interdisciplinary cooperation between technology experts, ethicists, and lawmakers will be essential to strike a balance between innovation and social responsibility.

Looking to the future, the continued advancement of AI holds immense promise, with the potential to revolutionize scientific research, enhance global sustainability efforts, and drive economic growth. Whether through accelerating medical discoveries, optimizing energy consumption, or advancing climate change mitigation strategies, AI has the capacity to tackle some of the world's most pressing challenges. However, its full potential can only be realized if it is developed and implemented in alignment with human values and societal needs.

Ultimately, the future of AI will depend not only on technological progress but also on the choices we make as a society. By fostering a responsible, inclusive, and forward-thinking approach to AI development, we can unlock new possibilities for human progress, ensuring that this powerful technology serves as a force for good, benefiting humanity as a whole.

## REFERENCES

- [1] D. B. Fogel, "Defining artificial intelligence," *Machine Learning and the City: Applications in Architecture and Urban Design*, pp. 91-120, 2022.
- [2] R. J. Silva-Jurado and M. D. Silva-Jurado, "Educational Innovation in the 21st Century: Gamification, Artificial Intelligence and Art as Transformative Tools," *YUYAY: Estrategias, Metodologías & Didácticas Educativas*, vol. 3, pp. 35-52, 2024.
- [3] N. Sfetcu, *Intelligence, from Natural Origins to Artificial Frontiers-Human Intelligence vs. Artificial Intelligence*: Nicolae Sfetcu, 2024.
- [4] W. Xu and F. Ouyang, "A systematic review of AI role in the educational system based on a proposed conceptual framework," *Education and Information Technologies*, vol. 27, pp. 4195-4223, 2022.
- [5] Y. Lu, "Artificial intelligence: a survey on evolution, models, applications and future trends," *Journal of Management Analytics*, vol. 6, pp. 1-29, 2019.
- [6] B. Shor and D. Schneidman-Duhovny, "Integrative modeling meets deep learning: Recent advances in modeling protein assemblies," *Current Opinion in Structural Biology*, vol. 87, p. 102841, 2024.
- [7] J. Jumper, R. Evans, A. Pritzel, T. Green, M. Figurnov, O. Ronneberger, K. Tunyasuvunakool, R. Bates, A. Židek, and A. Potapenko, "Highly accurate protein structure prediction with AlphaFold," *nature*, vol. 596, pp. 583-589, 2021.

- [8] A. Esteva, K. Chou, S. Yeung, N. Naik, A. Madani, A. Mottaghi, Y. Liu, E. Topol, J. Dean, and R. Socher, "Deep learning-enabled medical computer vision," *NPJ digital medicine*, vol. 4, p. 5, 2021.
- [9] S. M. McKinney, M. Sieniek, V. Godbole, J. Godwin, N. Antropova, H. Ashrafian, T. Back, M. Chesus, G. S. Corrado, and A. Darzi, "International evaluation of an AI system for breast cancer screening," *nature*, vol. 577, pp. 89-94, 2020.
- [10] A. Kamilaris and F. X. Prenafeta-Boldú, "Deep learning in agriculture: A survey," *Computers and electronics in agriculture*, vol. 147, pp. 70-90, 2018.
- [11] T. Wuest, D. Weimer, C. Irgens, and K.-D. Thoben, "Machine learning in manufacturing: advantages, challenges, and applications," *Production & Manufacturing Research*, vol. 4, pp. 23-45, 2016.
- [12] T. Ahmad, D. Zhang, C. Huang, H. Zhang, N. Dai, Y. Song, and H. Chen, "Artificial intelligence in sustainable energy industry: Status Quo, challenges and opportunities," *Journal of Cleaner Production*, vol. 289, p. 125834, 2021.
- [13] W. Liang, G. A. Tadesse, D. Ho, L. Fei-Fei, M. Zaharia, C. Zhang, and J. Zou, "Advances, challenges and opportunities in creating data for trustworthy AI," *Nature Machine Intelligence*, vol. 4, pp. 669-677, 2022.

# Heart Disease Prediction Using Machine Learning Models: A Comparative Study

Safa Gasmi  
Computer Science Departement  
Badji Mokhtar University  
LRI Laboratory  
Annaba, Algeria  
gasmisafa2@gmail.com

Sara Gasmi  
Computer Science Departement  
Badji Mokhtar University  
LRI Laboratory  
Annaba, Algeria  
gasmisara23@yahoo.fr

Akila Djebbar  
Computer Science Departement  
Badji Mokhtar University  
LRI Laboratory  
Annaba, Algeria  
aki\_djebbar@yahoo.fr

**Abstract**— heart disease remains one of the leading causes of mortality worldwide, highlighting the urgent need for accurate and efficient prediction models. In this study, we conduct a comparative analysis of several machine learning models to predict heart disease, including K-Nearest Neighbors (KNN), Support Vector Machine (SVM), Random Forest (RF), Extreme Gradient Boosting (XGBoost), and the Deep Forest model, specifically its cascade forest structure. The models were evaluated based on their performance metrics, with accuracy being the primary criterion. Our experimental results demonstrate that the Deep Forest model outperforms the other approaches, achieving the highest accuracy of 73.90%. This study emphasizes the potential of advanced ensemble methods like Deep Forest in medical diagnosis and underscores the importance of selecting robust models for heart disease prediction.

**Keywords**—Heart Disease Prediction, Machine Learning Models, Deep Forest, Medical Diagnosis

## I. INTRODUCTION

Heart disease is one of the most dangerous and widespread health issues worldwide, responsible for a significant number of deaths each year. Its impact on human life is profound, not only due to its high mortality rate but also because of the severe health complications it causes, such as heart attacks, strokes, and chronic heart failure [1]. The early and accurate prediction of heart disease is crucial for effective medical intervention, improving patient survival rates, and reducing healthcare costs. Traditional diagnostic methods often depend on clinical expertise and manual evaluation of patient data, which can be time-consuming, subjective, and prone to human error. Therefore, there is a growing need for more efficient and accurate predictive techniques [2].

In recent years, machine learning models have shown remarkable success in the medical field, offering innovative solutions for disease prediction and diagnosis [3]. These models are capable of analyzing large volumes of medical data, identifying intricate patterns, and making data-driven predictions with high accuracy. In the context of heart disease prediction, various machine learning algorithms have been employed, including K-Nearest Neighbors (KNN) [4], Support

Vector Machines (SVM) [5], Random Forest (RF) [6], and Extreme Gradient Boosting (XGBoost) [7].

Each of these models has demonstrated strong classification capabilities and unique strengths: KNN is simple and effective for pattern recognition, SVM excels in high-dimensional spaces, RF offers robust ensemble learning through decision trees, and XGBoost provides efficient gradient boosting for optimized performance.

Beyond these conventional approaches, advanced ensemble methods like the Deep Forest model [8] have emerged as promising tools for medical prediction tasks. The cascade structure of Deep Forest allows it to iteratively refine its predictions by passing feature representations through multiple layers of random forests, enhancing its ability to capture complex relationships within the data. This model's adaptability and ensemble nature make it particularly well-suited for the challenges of heart disease prediction.

The objective of this study is to conduct a comprehensive comparative analysis of these machine learning models, with a focus on identifying the most effective approach for heart disease prediction. By evaluating model performance on the dataset of heart disease most widely used, our aim is to determine the strengths and limitations of each method. This study underscores the potential of advanced ensemble techniques in medical diagnosis and highlights the importance of selecting robust and efficient models for heart disease prediction.

The remainder of this paper is organized as follows: Section 2 provides an overview of related work in the field of heart disease prediction using machine learning; Section 3 describes the methodology, including data preprocessing, model implementation, and evaluation metrics; Section 4 presents the experimental results and discussion; and Section 5 concludes with key findings and future research directions.

## II. Related work

Several studies have explored the application of machine learning models on large-scale heart disease datasets [9], particularly those comprising around 70,000 records,

demonstrating their effectiveness for this prediction task. For instance, Khan and Mondal [10] used various machine learning algorithms combined with feature selection techniques to estimate the risk of heart disease in patients. Among the tested models, SVM displayed a maximum accuracy of 72.22%.

Perva et al. [11] investigated the application of C4.5, KNN, and Naive Bayes algorithms with feature selection using WEKA tools. Their best result was achieved with KNN (K=53), reaching an accuracy of 73.05% on the heart disease dataset.

In their work, Maiga and Hungilo [12] implemented Random Forest, Naive Bayes, KNN, and Logistic Regression with Lasso regularization. The Random Forest algorithm performed best with an accuracy of 73% on the heart disease dataset containing 70,000 records.

USHA and KANCHANA [13] evaluated Logistic Regression, Decision Trees, AdaBoost, and Random Forest with Chi-square and Recursive Feature Elimination techniques. Their study showed that AdaBoost using Recursive Feature Elimination achieved 73% accuracy on the heart disease prediction task.

Jubier Ali et al. [14] applied Logistic Regression, Decision Trees, SVM, Naive Bayes, Random Forest, and KNN with ANOVA F-statistic for feature selection. Their Random Forest

implementation reached an accuracy of 69.41%, highlighting the challenges in improving prediction performance beyond certain thresholds.

Waigi et al. [15] conducted experiments on the Kaggle cardiovascular disease dataset using decision tree algorithms. Their optimized decision tree model achieved an accuracy of 72.77%, providing a good balance between model complexity and predictive performance for cardiovascular disease detection.

### III. MATERIAL AND METHOD

In this section, the proposed architecture for heart disease prediction is presented in Fig. 1. The objective of our approach is to conduct a comparative study of the most commonly used machine learning models. These models will be explained in the following sections.

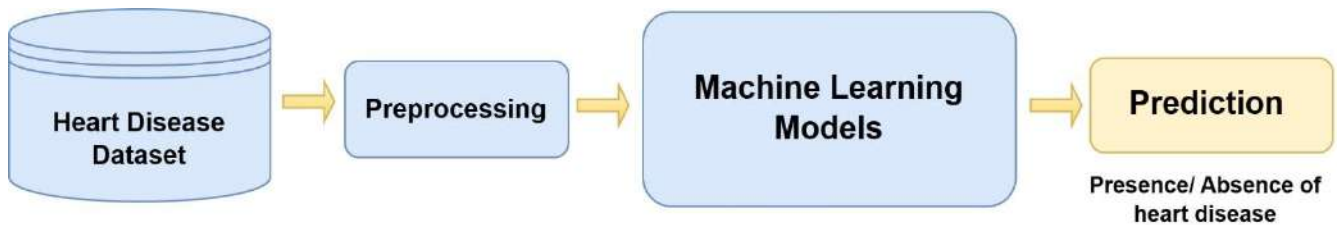


Fig. 1. The proposed framework for heart disease prediction.

#### A. Dataset description and Preprocessing

The Kaggle cardiovascular dataset consists of 70,000 patient records [16], each defined by 12 attributes. Fig. 2 illustrates the class distribution within this dataset, with 35,021 records assigned to the "heart disease" class and 34,979 to the "no heart disease" class, providing a balanced dataset suitable for training accurate diagnostic models. Table I presents a summary description of the 13 attributes in the CardioD dataset.

Before training the models, preprocessing steps are applied to ensure data quality and consistency. These steps include handling missing values, normalizing numerical features, encoding categorical variables. Additionally, duplicate records are removed to enhance model performance.

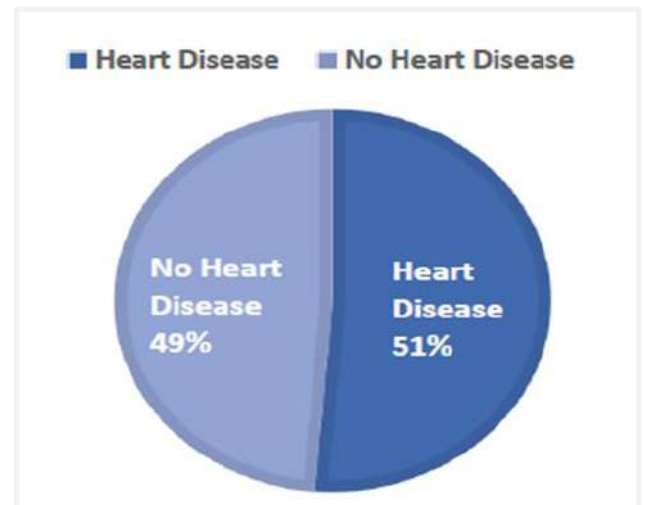


Fig. 2. Percentage distribution of classes in the dataset used.

TABLE I. BRIEF DESCRIPTION OF THE ATTRIBUTES OF THE DATASET USED.

Dataset	Attribut	Type	Description	Values
Heart Disease	age	Integer	Age of the patient (in years)	[30,65]
	gender	Categorical	0:female and 1:male	0-1
	height	Integer	Height of the patient in cm, from 55 to 250	[55,250]
	weight	Float	Weight of the patient in kg, from 10 to 200	[10,200]
	ap_hi	Integer	Systolic blood pressure, from -150 to 16020	[-150,16020]
	ap_lo	Integer	Diastolic blood pressure, from -70 to 11000	[-70,11000]
	cholesterol	Nominal	Cholesterol, there are three types which are: 1: normal, 2: higher, 3: well above normal	1-3
	gluc	Nominal	Glucose, there are three types which are: 1: normal, 2: higher, 3: much higher than normal	1-3
	smoke	Binary	1: smoking and 0: no smoking	0-1
	alco	Binary	1: alcoholic, 0: non-alcoholic	0-1
	active	Binary	1: does exercises, 0: does not exercise	0-1
	cardio	Binary	The diagnosis of heart disease, there are two classes which are: 0: does not have cardio, 1: has cardio	[3.5,298.7]

## B. Machine learning models used in heart disease prediction

Several machine learning models have been utilized for heart disease prediction. This study implements multiple models, which are described in the following sections.

### ✓ Support Vector Machine (SVM)

Support Vector Machine (SVM) is a supervised learning algorithm widely used for classification tasks, including heart disease prediction. It operates by finding an optimal hyperplane that best separates different classes in a high-dimensional space. The key principle behind SVM is margin maximization, where the algorithm seeks to maximize the distance between the hyperplane and the nearest data points, known as support vectors. To handle non-linearly separable data, SVM utilizes kernel functions such as linear, polynomial, and radial basis function (RBF) kernels, which map the data into a higher-dimensional space where a linear separation is possible. Despite its strong generalization ability, SVM can be computationally expensive, especially with large datasets, and requires careful tuning of parameters such as the regularization term (C) and kernel hyperparameters.

### ✓ K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) is a non-parametric algorithm that classifies a new data point based on the majority class of its K closest neighbors. The distance between data points is typically measured using Euclidean distance, although other metrics like Manhattan or Minkowski distances can also be used. KNN is particularly useful when the decision boundaries are complex and irregular. However, its performance is highly dependent on the choice of K: a small K may lead to overfitting, while a large K may smooth the decision boundary excessively, increasing bias. One of the major drawbacks of KNN is its computational inefficiency since it requires storing the entire dataset and computing distances for each prediction, making it less suitable for large-scale applications. Additionally, KNN is sensitive to the scale of features, necessitating proper feature normalization before training.

### ✓ Random Forest (RF)

Random Forest (RF) is an ensemble learning method that constructs multiple decision trees and aggregates their predictions to enhance accuracy and robustness. It operates by

training each tree on a randomly selected subset of the training data and randomly choosing a subset of features for each split,

which helps reduce overfitting and improves generalization. The final classification decision is made through majority voting among the individual trees, while regression tasks use the average prediction. RF is highly effective for structured data and handles missing values well. Moreover, it is less sensitive to noise compared to individual decision trees. However, RF can become computationally expensive when dealing with a large number of trees and may require hyperparameter tuning, such as adjusting the number of trees and the maximum depth, to optimize performance.

#### ✓ Extreme Gradient Boosting (XGBoost)

Extreme Gradient Boosting (XGBoost) is an advanced boosting algorithm that builds decision trees sequentially, where each new tree corrects the errors of the previous ones. Unlike traditional boosting methods, XGBoost incorporates several optimizations, including second-order gradient descent, regularization techniques (L1 and L2), and efficient handling of missing values, making it highly scalable and robust. It also employs a weighted boosting mechanism where misclassified instances receive higher importance in subsequent iterations, improving model accuracy. Due to its efficiency in handling structured data and its ability to capture complex patterns, XGBoost has become one of the most popular algorithms in machine learning competitions and real-world applications. However, it is prone to overfitting if the model is too deep or the number of boosting rounds is too high, requiring careful tuning of hyperparameters such as the learning rate, maximum tree depth, and the number of estimators.

#### ✓ Deep Forest (DF)

Deep Forest (DF) is a hierarchical ensemble learning approach that extends traditional random forests into a multi-layered structure. Unlike deep learning models that rely on backpropagation and large-scale data, Deep Forest iteratively refines feature representations by passing data through multiple layers of random forests and completely-random forests. Each layer extracts and transforms feature representations, progressively enhancing discrimination ability. One of the key advantages of Deep Forest is its adaptability to small and medium-sized datasets while maintaining strong predictive power. Additionally, it does not require extensive hyperparameter tuning and is more interpretable compared to deep neural networks. However, its performance depends on the depth of the cascade structure, and increasing the number of layers may lead to increased computational cost.

## IV. EVALUATION METRIC

To evaluate and compare the performance of the machine learning models used in this study for heart disease prediction, several metrics are employed, including accuracy, precision, F-score, and recall. Accuracy provides an overall assessment of correct predictions, while precision measures the proportion of correctly identified positive cases. The F-score offers a balanced evaluation by considering both precision and recall. These metrics are computed using the following equations:

$$\text{Accuracy} = \frac{TP + TN}{TP + FN + TN + FP} \quad (1)$$

TP: True Positive      FP: False Positive  
TN: True Negative      FN: False Negative

$$\text{Precision} = \frac{TP}{TP + FP} \quad (2)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (3)$$

$$\text{F-score} = 2 \times \left( \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \right) \quad (4)$$

## V. RESULTS AND DISCUSSION

Table II presents the performance evaluation of different machine learning models applied to heart disease prediction. The assessment relies on four key metrics: accuracy, precision, recall, and F1-score, which provide a comprehensive analysis of classification effectiveness. The obtained results highlight variations in model performance, with ensemble methods such as Deep Forest and XGBoost demonstrating higher accuracy and precision compared to traditional algorithms like KNN and SVM. A detailed comparison of these results is discussed below.

TABLE II. PERFORMANCE COMPARISON OF MACHINE LEARNING MODELS FOR HEART DISEASE PREDICTION.

Model	Accuracy%	Precision%	Recall%	F1-Score%
Random Forest	70.37	70.61	69.97	70.29
SVM	73.56	75.14	70.55	72.77
KNN	65.45	66.17	63.46	64.79
XGBoost	73.73	75.78	69.88	72.71
Deep Forest	<b>73.91</b>	<b>76.66</b>	68.87	72.56

The results indicate that ensemble-based models, particularly Deep Forest and XGBoost, achieved the best overall performance. Deep Forest obtained the highest accuracy (73.91%) and precision (76.66%), highlighting its ability to minimize false positive classifications. However, its recall (68.87%) was lower compared to SVM (70.55%), suggesting that SVM is slightly better at identifying actual positive cases.

XGBoost also demonstrated strong performance, with an accuracy of 73.73% and the highest F1-score (72.71%), indicating a balanced trade-off between precision and recall. In contrast, Random Forest showed slightly lower accuracy (70.37%) but maintained comparable recall (69.97%) to XGBoost.

SVM performed well in terms of precision (75.14%), effectively reducing false positives, but its recall (70.55%) was moderate. KNN, on the other hand, exhibited the lowest overall performance, with an accuracy of 65.45% and an F1-score of 64.79%. This suggests that KNN struggles with the complexity of heart disease classification, likely due to its sensitivity to data distribution and feature scaling.

Overall, the findings highlight the advantages of ensemble methods like Deep Forest and XGBoost in handling complex medical datasets. These models demonstrate a strong ability to generalize while maintaining a good balance between precision and recall, making them suitable candidates for heart disease prediction.

To illustrate the performance of the models in terms of accuracy, Fig. 3 provides a comparison of the obtained results.

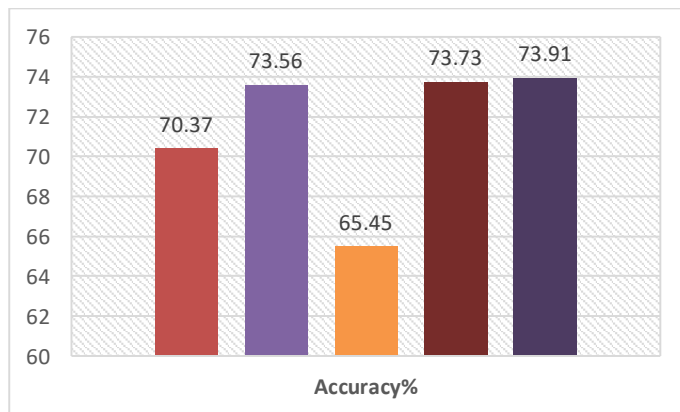


Fig. 3. Accuracy comparison of the applied machine learning models.

The following TABLE III presents a comparison between the proposed work and existing models in the literature for heart disease prediction.

Fig. 3. Accuracy comparison of the applied machine learning models.

TABLE III. COMPARISON WITH OTHER METHODS FOR HEART DISEASE PREDICTION

Ref	Method Used	Accuracy %
[10]	SVM, DT, KNN, LR, NB, RF	72.22
[11]	C4.5, KNN, Naive Bayes	
[12]	RF, Naïve Bayes, KNN, Logistic Regression	73 (RF)
[13]	Logistic Regression, decision Trees, AdaBoost, RF	73 (AdaBoost)
[14]	Logistic Regression, Decision Trees, SVM, Naïve Bayes, RF, KNN	69.41 (RF)
[15]	Decision Tree	72.77
<b>Our implemented models</b>	RF, SVM, KNN, XGBoost, Deep Forest	73.91 (Deep Forest)

The comparison highlights that the proposed approach achieves a higher accuracy (73.91%) compared to most existing studies. While previous works have explored various machine learning techniques with feature selection, their accuracy remains within a similar range, often not exceeding 73%. The Deep Forest model used in this study demonstrates a slight improvement over conventional models such as SVM, Random Forest, and KNN. This suggests that ensemble learning techniques, particularly Deep Forest, can enhance predictive performance for heart disease detection. However, further optimization and experimentation with additional feature engineering techniques could potentially refine these results even further.

## VI. CONCLUSION

In this study, a comparative analysis of various machine learning models for heart disease prediction was conducted. The results demonstrate that Deep Forest achieved the highest accuracy (73.91%), surpassing traditional models such as Random Forest, SVM, KNN, and XGBoost. This highlights the potential of ensemble learning techniques in improving predictive performance for medical diagnosis.

While existing studies have explored different feature selection methods and classification models, our approach provides a slight improvement in accuracy, suggesting that hybrid and advanced ensemble methods can enhance disease prediction capabilities. However, further research is needed to optimize model performance, particularly by integrating advanced feature engineering techniques and deep learning architectures.

Future work will focus on refining the proposed approach by incorporating additional medical datasets, exploring explainability techniques to enhance model transparency, and integrating domain-specific knowledge to improve clinical applicability.

## REFERENCES

- [1] S. Gasmi, A. Djebbar, and H. F. Merouani, "Boruta feature selection method for optimizing a case-based reasoning model to predict heart disease," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 37, p. 2351016, 2023.
- [2] S. Gasmi, A. Djebbar, and H. F. Merouani, "A Survey on Hybrid Case-Based Reasoning and Deep Learning Systems for Medical Data Classification," *Handbook of Research on Foundations and Applications of Intelligent Business Analytics*, pp. (pp. 113-141), DOI: 10.4018/978-1-7998-9016-4.ch006, 2022.
- [3] R. C. Deo, "Machine learning in medicine," *Circulation*, vol. 132, pp. 1920-1930, 2015.
- [4] M. Aci, C. Inan, and M. Avci, "A hybrid classification method of k nearest neighbor, Bayesian methods and genetic algorithm," *Expert Systems with Applications*, vol. 37, pp. 5061-5067, 2010.
- [5] M. A. Kumar and M. Gopal, "Least squares twin support vector machines for pattern classification," *Expert Systems with Applications*, vol. 36, pp. 7535-7543, 2009.
- [6] S. J. Rigatti, "Random forest," *Journal of Insurance Medicine*, vol. 47, pp. 31-39, 2017.
- [7] T. Chen, T. He, M. Benesty, V. Khotilovich, Y. Tang, H. Cho, K. Chen, R. Mitchell, I. Cano, and T. Zhou, "Xgboost: extreme gradient boosting," *R package version 0.4-2*, vol. 1, pp. 1-4, 2015.
- [8] Z.-H. Zhou and J. Feng, "Deep forest," *National science review*, vol. 6, pp. 74-86, 2019.
- [9] D. Shah, S. Patel, and S. K. Bharti, "Heart disease prediction using machine learning techniques," *SN Computer Science*, vol. 1, p. 345, 2020.
- [10] M. I. H. Khan and M. R. H. Mondal, "Data-driven diagnosis of heart disease," *Int. J. Comput. Appl.*, vol. 176, pp. 46-54, 2020.
- [11] F. Perva, H. Tucaković, M. Mušanović, and E. Yaman, "Prediction of cardiovascular disease," in *2022 XXVIII International Conference on Information, Communication and Automation Technologies (ICAT)*, 2022, pp. 1-5.
- [12] J. Maiga and G. G. Hungilo, "Comparison of machine learning models in prediction of cardiovascular disease using health record data," in *2019 international conference on informatics, multimedia, cyber and information system (ICIMCIS)*, 2019, pp. 45-48.
- [13] S. Usha and S. Kanchana, "Predicting heart disease using feature selection techniques based on data driven approach," *Webology*, vol. 18, pp. 97-108, 2021.
- [14] M. Jubier Ali, B. Chandra Das, S. Saha, A. A. Biswas, and P. Chakraborty, "A comparative study of machine learning algorithms to detect cardiovascular disease with feature selection method," in *Machine Intelligence and Data Science Applications: Proceedings of MIDAS 2021*, ed: Springer, 2022, pp. 573-586.
- [15] D. Waigi, D. S. Choudhary, D. P. Fulzele, and D. Mishra, "Predicting the risk of heart disease using advanced machine learning approach," *Eur. J. Mol. Clin. Med.*, vol. 7, pp. 1638-1645, 2020.
- [16] S. Ulianova, "Cardiovascular Disease dataset," 2019.

# Hybrid Adversarial Machine Learning for Robust Cybersecurity

1<sup>st</sup> Redjimi Mounira

Department of Computer Science,  
LIMA Laboratory, Faculty of Sciences  
& Technology, University of Chadli  
Bendjedid, El Tarf PB 73 36000,  
Algeria

[m.redjimi@univ-eltarf.dz](mailto:m.redjimi@univ-eltarf.dz) 

2<sup>nd</sup> Benmachiche Abdelmadjid

Department of Computer Science,  
LIMA Laboratory, Faculty of Sciences  
& Technology, University of Chadli  
Bendjedid, El Tarf PB 73 36000,  
Algeria

[benmachiche-abdelmadjid@univ-eltarf.dz](mailto:benmachiche-abdelmadjid@univ-eltarf.dz) 

3<sup>rd</sup> Maatallah Majda

Department of Computer Science,  
LIMA Laboratory, Faculty of Sciences  
& Technology, University of Chadli  
Bendjedid, El Tarf PB 73 36000,  
Algeria

[maatallah-majda@univ-eltarf.dz](mailto:maatallah-majda@univ-eltarf.dz) 

**Abstract**— The integration of Artificial Intelligence (AI) and Machine Learning (ML) has significantly transformed cybersecurity systems, particularly in malware detection. However, Deep Learning (DL) models remain highly vulnerable to adversarial attacks such as data poisoning, backdooring, and evasion through subtle perturbations. Traditional static and dynamic defense mechanisms are increasingly ineffective against adversarial malware specifically designed to deceive learning-based classifiers. This research addresses these challenges by focusing on adversarial threats that combine randomness with knowledge of the target system. The study explores two key defensive strategies: adversarial training and defensive distillation. Adversarial training enhances model robustness by incorporating adversarial examples during the learning process. Defensive distillation improves resistance to attacks by smoothing decision boundaries through knowledge transfer between neural networks. A Hybrid Adversarial Machine Learning (HAL) framework is proposed, integrating both techniques into a two-phase defense strategy. Experimental evaluations on public malware datasets, including GAMs, EMBER, and Maling, demonstrate that the hybrid approach outperforms existing defense methods in terms of robustness and detection accuracy. The results highlight the effectiveness of HAL in strengthening cybersecurity systems and emphasize the importance of continuous model hardening in security-critical environments.

**Keywords**— Adversarial Machine Learning, Cybersecurity, Malware Detection, Adversarial Training, Defensive Distillation, Hybrid Learning Models, Deep Learning (DL), Adversarial Attacks, Machine Learning (ML), Intrusion Detection Systems

## I. INTRODUCTION

The unprecedented advancement in Artificial Intelligence (AI) and Machine Learning (ML) technologies has fundamentally changed the way technology integrates into our life in recent years, empowerment of AI-ML applications ranging from multimedia online app to traffic management system leading to smart cities that augurs the significance of the paradigm shift to achieve sustainable developments in education, health, commerce, social, and governance. However, the increasing sensitivity of the information processed in the AI-ML technologies coupled with growing cyber threats has raised security and privacy concerns prompting megalith investment by global states and seizure of power to gain the upper hand propelled by AI arms race. The preminent application and disclosure of deep learning (DL) models made them susceptible to a slew of adversarial

cyber threats like backdooring based on poisoning training data, stealing training data, fabricating biased and malicious predictions, extracting sensitive attributes about the data, fooling predictions by slight perturbations, extracting intermediate information or input samples or incorrect predictions or confidence scores etc. [1]. Such a clever hack either by exploiting the vulnerabilities specific to the mode of operation, or technology agnostic approaches, cripple the entire system eliciting the demand for an integrated platform of operation agnostic defense mechanism.

Transgressing the conventional malware detection and filtering methods that are ill-suited for the impregnation of ML models in the cyberbattle ground is crucial to combat these crusader threats. Recent endeavors in understanding the source of vulnerability and eliciting solution mechanisms at the input and model level along with proposing multiple fast-attack paradigms and transformations has given an impetus to a growing new domain of robust adversarial ML as an innovative approach to cyberthreats. A thorough review of the holistic adversarial robustness of ML model puts forward a clear guideline for the safe and secure applications of the technology across all domains [2].

The rest of the paper is organized as follows: Section II introduces the foundational concepts of machine learning in cybersecurity and provides a comprehensive review of adversarial attacks and defenses, with a particular focus on malware detection systems. Section III discusses adversarial training techniques, while Section IV presents defensive distillation as a complementary robustness mechanism against adversarial perturbations. Section V introduces the proposed Hybrid Adversarial Machine Learning (HAL) framework, detailing the integration of adversarial training and defensive distillation, along with its benefits and associated challenges. Section VI outlines the research methodology, including dataset construction, model development, training strategies, and evaluation metrics. Section VII reports and analyzes the experimental results, highlighting the robustness and scalability of the proposed approach through comparative performance evaluations. Section VIII discusses the theoretical and practical implications of the findings, followed by Section IX, which outlines ongoing research activities and future research directions in the context of evolving adversarial threats and large-scale cybersecurity systems. Finally, Section X concludes the paper by summarizing the main contributions

and emphasizing the necessity of robust and resilient machine learning models for modern cybersecurity applications.

### *1. Background and Significance*

Non-traditional computing platforms such as quantum computers and DNA-based computers have been studied for decades with application in a variety of fields like cryptography, artificial intelligence, etc. Exploration of quantum properties in quantum-dot cellular automata has uncovered possibilities for applications in various fields. Quantum-dot cellular automata are a set of potential candidates for implementing non-traditional computing. Hybrid systems made of two different types of nano-components can combine their individual strengths to outperform the computing systems made of similar components. Quantum-dot cellular automata-laser diode hybrid has been proposed for applications in ultra-low power nano-scale classical logic gates. Similarly, quantum-dot cellular automata-magnetic tunnel junction hybrid systems have been explored for their future use in spintronic computing systems. Hybrid quantum-dot cellular automata-superconductor nano-wire system can power low-threshold Josephson logic devices. Nanostructures such as quantum-dot cellular automata, quantum-dot cellular automata-wires, and quantum-dot switched quantum-dot cellular automata reviewed as candidates for high-speed on-chip interconnects for future nano-scale circuits [3]. However, implementation of quantum-dot cellular automata has remained a challenge due to the high complexity of the manufacturing process.

In this work, a new kind of quantum-dot cellular automata-nano-electromechanical system has been investigated. A potential unit cell of quantum-dot cellular automata-nano-electromechanical system has been presented. First-principal models including electrostatic and piezoelectric effects, thermodynamic modelling, mathematical modelling of current industrial processes, etc., were developed. Computationally efficient methods that enhance the energy-efficiency and the speed of the quantum-dot cellular automata-nano-electromechanical unit cells by several orders of magnitude via electro-mechanical amplification of the feed-forward signals and modelling the uncertainties, respectively have been proposed.

### *2. Research Problem and Motivation*

Cybersecurity faces many challenges stemming from the rapid proliferation of cyber threats, new attack vectors, and adversarial attacks. To mitigate the impacts and reduce harm from malware incidents, malware detection systems are widely adopted to find potential attack signatures and stop the execution of suspicious files [4]. However, malware detection technologies suffer from new threats that have been specially crafted to deceive machine learning/convolutional mechanisms. These crafted threats are termed adversarial malware, and attacks that deliberately generate and deploy these adversarial malware instances are called adversarial attacks.

This research addresses the challenges of currently deployed static and dynamic cybersecurity measures in protecting systems and networks against adversarial malware attacks. Many of the currently system and network-based cybersecurity measures, such as firewalls and intrusion

detection systems, use rule-based mechanisms for the detection and mitigation of malware threats. They suffer from many limitations and are vulnerable to many static cyber threats. In addition, as malware attacks evolve and become more advanced through malware polymorphism and metamorphism, current cybersecurity measures have significantly reduced effectiveness for the detection of new and unseen variants of known malware families. Also, with the advent of deep learning in detecting malware, deployed malware detection systems are vulnerable to adversarial malware inputs, which are malicious inputs specially crafted to deceive and mislead malware classifiers. Input-dependent adversarial malware attacks have been found to be more effective and evasive against malware classifiers trained with deeper architectures [1].

### *3. Research Objectives*

The need to advance cybersecurity for vital infrastructures has been one of the main objectives of the Department of Homeland Security. Cybersecurity has focused on protecting hardware, software, and information systems from physical, network, or digital attacks, in addition to malicious events caused by unintended errors. There is increasing reliance on system approaches that form the basis for modern daily technologies such as transport, public health, energy, defense, and information technology. Nonetheless, the effects of malicious disruptive design events are hardly understood and currently without a protection framework. This research proposal presents a pioneering approach that tries to understand, classify, and gain resilience against disruptive attacks in system approaches today [2].

There is a need to comprehend arbitrary hybrid attacks that combine randomness and knowledge of the target system. The aim is to create theoretical and experimental design examples that will clarify attacking processes and the vulnerabilities of targeted systems. In conjunction with experimental studies, a systematic development of designs, their robustness, and defense against such attacks is of prime importance. The objective is to develop a new scientific research field and provide a formal framework, methods, and design tools, paving the way for major advancements in innovative engineering design of complex systems with controlled behavior and increased resilience.

## **II. FOUNDATIONS OF MACHINE LEARNING IN CYBERSECURITY**

This section serves as an educational foundation. It begins with a general overview of machine learning. The definitions of supervised, non-supervised, and reinforcement learning are conveyed. Some commonly used machine learning models are briefly introduced. These building blocks of machine learning are provided specifically in the context of cybersecurity. This is followed by a sub-section that describes adversarial attacks in the machine learning domain. Recent developments of adversarial attacks in non-image domains, such as the cybersecurity domain, are extensively surveyed. This section covers various types of vulnerabilities in cybersecurity machine learning systems against evasion, poisoning, and other types of attacks. Any prerequisite

knowledge regarding machine learning and cybersecurity is provided to ensure coherence of the subsequent sections.

Machine learning (ML) refers to the automation of statistical learning tasks performed by a machine. By statistically analyzing the past data collected, a machine can perform tasks and make decisions. In other words, it allows a machine to learn for itself based on previous data, without needing explicitly programmed rules. ML can be regarded as a computer program that can be run on a computer to achieve a task. The common methods for achieving these tasks are regression and classification. In regression tasks, a mapping function is learned from input variables to output continuous values. On the other hand, in classification tasks, given some historical data, a decision function is learned after analyzing the common properties of different data samples. This function is further used to classify a new sample into one of the different classes it belongs to. Classic examples of ML-based classification tasks include email filtering (spamming or not spamming), sentiment analysis (negative, neutral, or positive), and accident attribution (blame or not blame) [5].

There are three types of conventional learning setups in ML. They are known as supervised learning, unsupervised learning, and reinforcement learning. In supervised learning, a model is learned from training data containing input-output pairs in which the output classes are known. In non-supervised learning, the model is trained using unlabeled data, in which the output labels are unknown. As a result, the model must learn to discover the underlying patterns of the examined samples. Finally, in reinforcement learning, an agent learns the optimal setup of an environment using feedbacks of performed actions, which can be either rewards or penalties [6].

### 1. Machine Learning Basics

The foundations of machine learning relevant to cybersecurity are presented. Machine Learning (ML) refers to a collection of technologies capable of learning unknown phenomena from data [5]. It is divided into three categories: supervised, semi-supervised, and unsupervised learning.

In supervised learning, all used data is labeled with the output of interest, such as a class label representing a certain kind of malware. In semi-supervised learning, only part of the data is labeled, while in unsupervised learning, all data is unlabeled. The goal of learning models is to find a mapping from the input data space to the output space of interest. This mapping is built from a set of labeled data called the training dataset. Models can take different forms, such as a deterministic mapping or a more general probabilistic modeling respecting the uncertainty over the data [7]. The mapping is adjusted by minimizing a loss function that describes the optimality of the obtained mapping. The output of this function should be as small as possible for an excellent mapping approximation.

The final model is used to make predictions on unseen data to understand its generalization capabilities, which is the

ability to classify data outside the training data used. The unseen input is called a test dataset, containing data from the same distribution as the training one but with new examples. The predicted output from the ML model is compared with the actual test data output, which is usually known, allowing the quantification of the model's performance. Standard measures to evaluate model performance include computing the number of misclassifications and quantifying performance in terms of the true positive rate, false positive rate, precision, F1-score, and area under the ROC curve (AUROC).

### 2. Adversarial Attacks in Machine Learning

To develop a hybrid adversarial machine learning model, it is important to first understand the many different attacks done to machine learning models that lead to compromised cybersecurity. There are many ways and strategies to attack a machine learning model, especially student models. Some commonplace adversarial attacks include data poisoning, model poisoning, evasion, stealthy evasion, inference and membership attacks, and supply chain attacks. Data poisoning attacks attempt to introduce poisoning samples into the training set, thereby misleading the learning process of the ML model(s) [8]. Many data poisoning attacks suggest dropping samples belonging to targeted classes, and adding (untargeted) malicious samples close to the chosen target input class, thereby enhancing the model's confusion towards the output of the target input class. These attacks are devastating as they span many attacks and exist in many commonly used datasets including CIFAR-10 and MNIST. Model poisoning attacks attempt to manipulate the output of the learned model using a few poisoned samples and work is being done to evaluate a robust aggregation algorithm for federated learning under such model poisoning attacks. Most recent systematization of evasion attacks in the context of ML-based cybersecurity defense is done. It is stated that most evasion attacks have various attacks on input feature space, including by-passing the network and application specific attack models. Such attack examples include using adversarial perturbations of images, crafted Office or PDF documents, spreading forged but harmless messages in social networks, and creating a pool of resources for a sophisticated DDoS attack.

In addition to the many different front-running attack methods, there are also a plethora of different layers of knowledge and complexity when it comes to current relentless attacks on models. A white-box attack includes a complete knowledge of attack policies such as the model architecture, target and output parameters. For example, if group A attempts a white-box attack on the model user group B, A can simulate the learning process of model B and, using the output of the user model, generate adversarial samples. A black-box attack involves a complete lack of knowledge of the target model policy. Using the model output to probe about the decision boundary of the model, one adversarial example mislead model A is likely to mislead model B. Adversarial attacks on deep learning models are generally divided into targeted and untargeted attacks. There are many other attacks on both the side of knowledge of the attacker to a model and the different scope of the attacks [9].

### III. ADVERSARIAL TRAINING

Adversarial training is a concept within the field of machine learning whereby a model is explicitly trained on adversarial examples, either to make the model robust to attack or to reduce its test error on clean inputs. Adversarial examples are observably different from the original inputs but are usually produced by modifying them and are therefore very close to the original inputs [10]. Adding appropriate perturbations to adversarial examples makes them fail to be classified correctly by a machine learning model, such as a Convolutional Neural Network (CNN), with a high level of confidence. Perturbations are often termed adversarial noise and can include simple techniques like white noise, impulse noise, Gaussian noise, etc., or be generated by sophisticated techniques such as Fast Gradient Sign Method (FGSM) or Carlini–Wagner Attack (C&W).

In cybersecurity, adversarial training is the process of utilizing adversarial examples to train a machine learning model within a protection system or intrusion detection system (IDS), therefore increasing the chance of rejecting or preventing attacks. Adversarial examples can also be produced based on the machine learning model that is employed in the protection system. So, adversarial training allows the protection system to be aware of and trained against diversely sophisticated attacks. By training with different adversarial examples, the model incorporates knowledge of those attacks, making it increasingly challenging for adversaries to deceive it. As a result, the robustness of cybersecurity systems can be enhanced against adversarial attack techniques with a variety of strategies [11].

#### 1. Concept and Principles

Adversarial training is founded on the principle of proactively strengthening machine learning models by exposing them to adversarial examples during the learning process. Rather than relying solely on clean training data, the model is trained on a mixture of legitimate and adversarial perturbed samples, allowing it to learn more robust decision boundaries and reduce sensitivity to malicious input manipulations. According to the holistic adversarial robustness framework discussed in [2], incorporating adversarial samples directly into the training phase is one of the most effective defenses against evasion and poisoning attacks, as it aligns the learning objective with worst-case threat scenarios.

From a theoretical perspective, adversarial training can be viewed as a min–max optimization problem, where the model seeks to minimize classification loss while simultaneously accounting for the maximum perturbation an adversary can introduce within a defined constraint space. As demonstrated in large-scale adversarial learning studies [10], this principle enables models to generalize better under adversarial conditions by explicitly modeling the attacker's behavior. In cybersecurity contexts, this paradigm is particularly relevant, as adversarial malware often exploits subtle feature-level perturbations to bypass detection systems. Consequently, adversarial training establishes a foundational defense mechanism by

embedding attack awareness into the learning process itself, thereby enhancing the robustness and reliability of ML-based cybersecurity solutions.

#### 2. Techniques and Algorithms

The AI solutions employed in cybersecurity and privacy protection domains generally consist of machine learning as well as deep learning algorithms. The algorithms that operate in the aforementioned domains are subject to adversarial attacks. These attacks are crafted in such a way that they evade detection but still resemble a legitimate program. The design of such attacks is challenging and, in many cases, requires collaboration with a cybersecurity analyst or the knowledge of proprietary information. On the other hand, adversarial training or defensive distillation refines trained classifiers models to improve resilience against future attacks on artificial intelligence systems [12].

Typically, the adversarial training procedure involves a) data collection of the pre-classification steps such as API calls or network flows, b) selective generation of instances representing intra-class overlapping for each candidate model, c) model retraining with augmented dataset consisting of additional adversarial instances. For adversarial training to be effective, it is essential that different obfuscation algorithms are utilized at each iteration of training and testing. States of the art obfuscating, detection and classification models are utilized and novel techniques are designed to enhance their robustness [10].

Considering the complexity of modern pipelining systems, the proposed solution is suitably hybrid. Indications of simplistic internal parameters failing even conceptually or academic research dressing a complex problem with oversimplifying methodology have been illustrated. It is therefore expected that the hybrid approach will result in a combination of mechanisms and techniques that will improve resilience against adversarial machine learning attacks.

#### 3. Applications in Cybersecurity

Adversarial training has several practical applications in real-world cybersecurity scenarios. One such application involved Dr. Ben of Security Scorecard, an independent cybersecurity ratings company that provided the Security Scorecard platform—an artificial intelligence (AI) system for monitoring data breaches and malware outbreaks. Even small amounts of compromised data, referred to as "leaks," could potentially harm the company's customers if they were not discovered and remediated quickly. Such leaks could allow malicious hackers to use the stolen information to impersonate, blackmail, or otherwise compromise those individuals, which could pose a serious liability for Security Scorecard. The AI system was created and trained using historic leak data, using Machine Learning Classifiers (MLCs). Because these leaks were not observed by humans, the AI system had to generalize from weak signals. For instance, some email address leaks might provide only a username, whereas other leaks would provide additional information such as passwords, home addresses, and birth dates. The AI system was able to construct data models

from these signals and assign company "scores" which captured suspicion/hacks [13].

Another application in the financial industry was implemented to enhance fraud detection for a banking services provider. The sensitive nature of such data made it impossible to either share or build on the training environment; thus, adversarial training was effectively used to simulate hypothetical fraud attempts and build a generalized fraud detection system. Another example involved a healthcare provider, whereby data was partitioned both horizontally and vertically to secure sensitive patient data. A deep-learning network was created on top of the federated architecture to allow medical image data sharing while still maintaining user privacy. This architecture received the first prize at the Data Sharing & Security Award of the 2020 Intel AI Global Impact Festival [14].

#### IV. DEFENSIVE DISTILLATION

In Modern Deep Neural Networks (DNNs), input feature perturbations can significantly affect the overall outputs, motivating the definition of Adversarial Examples (AEs) within a given epsilon-ball. In prior works, AEs were found to be imperceptible to human eyes. Carlini and Wagner crafted sophisticated, norm-based DNN adversarial examples that are imperceptible and robust to DNN defenses for large scale image datasets [15]. In contrast, Paper not et al. attacked DNNs by approximating DNNs with smaller models via knowledge extraction, which can be used to craft transferable adversarial examples [16].

On the defense side, several techniques against the above attacks have been proposed. For example, training DNNs with adversarial examples has been shown to enhance model resilience on the original input distribution, but not against out-of-distribution samples with similar perturbative margins. The most related work to this effort is Defensive Distillation, which was viewed as a way to increase SoftMax temperature in the cross-entropy loss function for training. To increase classification robustness, an exactly the same model is distilled with a larger temperature parameter, resulting in softer class probabilities. The knowledge distillation framework is based on the fact that for highly accurate pretrained DNNs, SoftMax outputs on the same input follow a specific distribution induced by the model and data. Under this distribution, a student DNN with shared architecture will learn to behave like the teacher when mimicking the output probability distribution.

##### 1. Concept and Principles

Defensive distillation is an approach by which a neural network (NN) can be modified to become more resistant to adversarial input. In addition to consideration of the neural net's output class predictions, the separation of its hidden layer becomes an emerging possible explanation of adversarial examples, and one that defensive distillation attempts to exploit [16]. A distilled network is either the same network architecture trained with a high temperature SoftMax cross-entropy in addition to the traditional loss, or

it can be an entirely new architecture trained on the soft targets produced by the first network.

Cyber-Physical Systems (CPS), for example, use a multitude of different sensors and actuators to monitor an environment and interact with it. Traditional Networked Control Systems (NCS) made limited use of the network to transfer digital data, while nowadays, the whole computational side of the systems is likely to be transferred to the cyber component. With the introduction of networking and computation structures, the CPS vulnerabilities are dramatically increased. Control system crimes include intruders taking over control of a plant to cause illness or death resulting from incorrect drug dosages, bank robbery emergencies like turning off an ATM alarm, or physically damaging year-long research machinery. Therefore, preventing tampering with a fully abstract representation of reality is a first step in making these systems safer [15].

##### 2. Techniques and Algorithms

[16]. Model distillation relies on an auxiliary model, typically simpler or a smaller version of the pre-trained model. The input data is shifted through both the original model and the auxiliary model. The outputs are then trained via a cross-entropy loss function, where the logits of the SoftMax layer of the models act as input. As an output, the auxiliary model captures the knowledge of the pre-trained deep neural network. In the context of model distillation as a defensive technique, an attacker first trains a model targeting a predefined original model, use the improbable deeper SoftMax layer to craft the adversarial samples, and later feeding the new samples to the auxiliary model trained with a cross-entropy loss function.

The notion of security games is introduced, which considers hybrid settings where both a defender and an attacker can recognize ML classifiers as in the closed-world hypothesis, but with some reasonable assumptions regarding the use of such classifiers. The attacker possesses a pre-trained model, a target that a defender tries to deceive, and knowledge of the defender strategy, while the defender possesses access to the model architecture and data but not its parameters. Pseudo class attacks exploit the usage of the class output of a pre-trained alternative model to attack another with the same architecture that runs on the same underlying data [15].

##### 3. Applications in Cybersecurity

###### Defensive Distillation

Deep neural networks (DNNs) have been used in several cybersecurity applications, such as network intrusion detection and malware classification. As DNNs are frequently deployed in security-critical problems, maintaining their correctness and integrity when subjected to malicious actions is vital. Adversarial examples, or samples intentionally designed to mislead classifiers, have been shown to successfully bypass learning-based intrusion detection systems [16]. The adversarial perturbations that are minimally perceptible to human observers are crafted to deceive machine learning algorithms. Therefore, understanding these perturbations and developing robust

classifiers is crucial for the long-term success of intelligent automated systems in cyber environments. These classifiers must be built robust to previously unseen perturbations to enhance the overall integrity of the system.

Defensive distillation, based on knowledge distillation, is proposed to defend against adversarial attacks. It consists of two steps: first, a teacher DNN is trained on the original dataset; then a student DNN is trained on the soft labels generated by the teacher. The output layer is replaced by a SoftMax function that employs a temperature parameter to smooth the predicted probability distribution. Out of the two, only the distilled DNN is used in the classification process. The rationale behind this approach is that deep network models trained by greater temperature values yield more informative probability distributions. Defensive distillation reduces the effectiveness of the most influential gradient-based attack technique. DNNs can be securely deployed to protect against adversarial sample crafting in various real-world applications such as malware classification.

## V. HYBRID ADVERSARIAL MACHINE LEARNING

There are two ingenious ideas proposed in two seminal papers. Both approaches alleviate DNNs susceptibility against adversarial samples and thus are thought to complement each other against separate classes of attacks. However, to the author's best knowledge, the combined approach has never been investigated or discussed in the context of adversarial machine learning.

In the first approach [16], the objective is to precisely capture the behavior of the DNN under expected input conditions and then use this information to train a more robust model. This is achieved with a two-step process in which a new model is trained to mimic the behavior of the DNN. The DNN's outputs are modified using a SoftMax temperature parameter  $t$ . High-temperature SoftMax functions generate more informative predictions by favoring low-probability labels while distinguishing between the majority of samples which are predicted with high confidence to the same label. Lowering  $t$  results in more "confident" predictions, as handle the problem one class at a time, rather than all labels at once.

In the second approach, adversarial training is performed to augment the training set with adversarial samples, thus achieving robustness on test examples. A robustness against adversarial samples is gained because the attack is defeated by adding an adversarial sample for which the objective function output is largest. A trade-off is introduced between secure (low-prevalence) and optimal (low risk) classifiers, allowing desired DSS to be ensured.

### 1. Integration of Adversarial Training and Defensive Distillation

[16]

The seamless integration of adversarial training and defensive distillation within the cybersecurity framework is introduced here. Two distinct approaches to improving the robustness of machine learning models against adversarial

attacks are taken. Each approach is described in detail, followed by an exposition of their architecture and operational integration. Adversarial training is designed to be the first phase of defensive distillation's robustness improvement scheme and builds upon the original architecture proposed by. Defensive distillation is the second phase of adversarial training and is operated as a separate step subsequent to the original adversarial training and innocence of the threat model.

### 2. Benefits and Challenges

The hybrid approach offers several benefits for enhancing cybersecurity. Firstly, it improves the robustness of machine learning models against adversarial attacks by combining the strengths of different models and generating diverse adversarial examples. This diversity makes it harder for attackers to craft effective attacks that can evade detection by all models in the ensemble [1]. Secondly, it enhances the models' ability to detect and classify unknown attacks, including previously unseen types of malware or intrusion attempts. By learning from a variety of models and attack scenarios, the hybrid system can better generalize to new threats and reduce the risk of false negatives [13]. Thirdly, it increases the transparency and interpretability of machine learning-based cybersecurity solutions by providing insights into how different models make decisions and how they respond to different types of attacks. This transparency can help security analysts understand and trust the system's predictions and take appropriate actions.

However, several challenges need to be addressed to ensure the effectiveness and feasibility of a hybrid approach in real-world applications. Firstly, the complexity and resource requirements of the hybrid system may increase significantly, as it involves training and maintaining multiple models, generating and storing multiple adversarial examples, and integrating the predictions of different models. This complexity may lead to higher computational costs and slower response times, which are critical factors in cybersecurity. Secondly, the effectiveness of the hybrid approach depends on the diversity and complementarity of the machine learning models and the adversarial examples used. If the models are too similar or the examples are too generic, the benefits of the hybrid system may be limited. Thirdly, the security of the hybrid system itself needs to be considered, as attackers may attempt to target multiple models or find adversarial examples that can fool the ensemble.

## VI. RESEARCH METHODOLOGY

The approach taken in this study to mitigating adversarial attacks with and against ensemble classifiers is straightforward in principle, addressing the main limitations of existing work. Most importantly, it extends to classifier ensembles composed of arbitrary base classifiers, as all attacks independently manipulate the inputs to the classifiers originally considered, while the defenses in general brings the model variations together. This research is composed of one attack (Hybrid) and one corresponding defense (Ensemble + Stacked). Both the attack and the

defense are examined in regard to different strategies for choosing the constituent base classifiers. The experimental results validate the models and indicate that the base classifiers significantly impact the performance of aggressive targeted adversarial attacks and the computational cost involved [4]. Details of the model development, data preparation and collection, model training, and evaluation metrics are provided.

**Datasets.** To verify the generality of Hybrid, two public cross-dataset settings are addressed and three public datasets are used: the Microsoft Malware Classification Challenge dataset (GAMs), the EMBER dataset, and the Maling dataset [17]. The datasets are publicly available and designed for different anomaly-based malware research. Each dataset contains labeled images or vectors of either benign or malicious samples of different initial Android apps. Further explanations on each dataset are provided, and sample benign and malicious images from the datasets used are illustrated.

### 1. Data Collection and Preparation

The dataset is constructed from two sources. Firstly, a raw dataset containing cyber-attacks is developed by conducting extensive network-level cyber-attacks experiments on various systems, comprising a reference dataset of benign traffic, and datasets for command and control, denial of service, web attack, and reconnaissance attack classes [18]. The second source employs benign traffic and malware activity on the Windows OS from the generated EMBER dataset and the malware collection dataset taken from URLs provided in GitHub repositories. The activity is logged and analyzed in a raw format in csv files. The cap files containing the traffic flows from both datasets are generated using the Wireshark tool. Upon scrutinizing the traffic flows with malware activity, several distinct and relevant features are extracted.

The features are selected based on the consideration of their relevance to malware activity and varying characteristics under attack and benign conditions. A total of fifty-five features from the two datasets are selected, comprising one character feature and fifty-four numerical features [4]. When a specific feature is considered, an individual numerical feature for the traffic flow containing that feature is derived. When multiple features are considered, all the numerical features for the traffic flows containing any of those features are derived. These are classified as CWE (Common Weakness Enumeration), SC (Security Classification), IP (Internet Protocol), FQDN (Fully Qualified Domain Name) and DNS (Domain Name System) character features, and various numerical encoding for the remainder features.

### 2. Model Development and Training

Model Development and Training outlines the approach taken to create the machine learning models used throughout the study. It details the selection of the initial model architecture, the methodologies employed for training and refinement, as well as the implementation of parallel training approaches.

The initial model architecture developed for this research was a variation on the convolutional neural network architecture LSD [7], which was designed around a malware dataset of disassembled portable executable (PE) files from the Microsoft Windows operating system. The feature extraction pre-processing for this dataset was designated as the modelling baseline, and computational resource requirements were assessed. Implementations in the programming languages Lua and Python were developed. The Lua implementation was optimized to execute on Graphical Processing Units (GPUs), while the Python implementation used TensorFlow and Keras on CPUs. All models trained on this initial architecture executed on CPUs due to limitations in GPU availability, thus the additional computational resource requirements were prohibitive to complete model training in reasonable timeframes. The initial training for the Python implementation progressed for 22 hours until the epoch 34 model was saved, which left nearly the entire training set unexplored, so model performance was still near the baseline accuracy of 0.69.

In an ongoing effort to reduce training timeframes, a parallelized alternative architecture with a simplified feature extraction process was considered. The goal was to develop a model with a smaller architecture that could nonetheless perform sufficiently to explore poisoning conditions, thus requiring fewer resources and longer training timeframes. Simultaneously, the parallel training approach generated a variety of models that did not utilize CPU parallelization and executed on CPUs instead. Overall, models were developed and assessed using a variety of architectures, learning methods, and immigration strategies [2].

The selected model architectures, training methodologies, and pre-trained model representations used throughout the study were developed and described in Version 1 of the modelling, learning and generation frameworks. Ma3L is a publicly available open-source machine learning framework for malware classification and mitigation model development, training, and assessment under adversarial conditions, capable of both parallel and standalone execution. The framework has been packaged as a Docker image for flexibility in execution options and to accommodate varying machine configurations. Parallelized versions of the models in training frameworks are also provided for off-the-shelf execution, only requiring the compilation of Ma3L.

### 3. Evaluation Metrics

This section discusses the evaluation metrics used in the research to assess the performance and effectiveness of the machine learning models in the context of cybersecurity. With the rise of sophisticated machine learning cyber-attacks, it is important to evaluate the robustness and reliability of the machine learning models. This research aims to investigate the use of hybrid distant based detection techniques to enhance the robustness of machine learning models and evaluate their performance using a variety of metrics.

The evaluation metrics employed in this research include Accuracy, Precision, Recall, F-Score, and False Positive Rate: - Accuracy: Accuracy measures the percentage of correct predictions made by the Machine Learning model in relation to the total number of predictions. - Precision: Precision quantifies the proportion of true positive predictions out of all positive predictions made by the model. It is an important metric when the cost of false positives is high. - Recall: Recall, also known as sensitivity or true positive rate, measures the proportion of true positive predictions out of all actual positive instances. It is important when the cost of false negatives is high. - F-Score: The F-score, or F1-score, is the harmonic mean of precision and recall, providing a single score that balances both metrics. - False Positive Rate: The false positive rate calculates the proportion of negative instances that are incorrectly classified as positive by the model [4].

## VII. EXPERIMENTAL RESULTS AND ANALYSIS

The experimental results and analysis derived from the study are presented in this section. Hybrid Adversarial Learning (HAL), consisting of Generative Adversarial Learning (GAL) and Defended Adversarial Learning (DAL), is trained on each malware family using a Google laptop with an Intel(R) Core (TM) i5-6260U CPU clocked at 2.00 gigahertz (GHz) and an Nvidia GeForce 940M GPU card. All models' training begins with two epochs without adversarial training to allow the Knowledge Transfer step to work properly. Knowledge Transfer (KT) starts after the two epochs. The PGD attack is used with 40 steps, each step changing the model inputs by 0.05, and both the early stopping and loss thresholds are set to the default values of 10 epochs and 0.003, respectively. It should also be mentioned that the batch size is set to 64, and all training is done with a scaling parameter for the GAN's loss of 1.0. Adversarial weight perturbation (AWP) is also adapted here for an SDL application.

HAL's performance against the individual adversarial attack is compared. The experimental results are divided into three parts: the first part shows HAL's performance across various components in a complete malware detection framework (17.2.1), the second part looks into the issues of shared defenses (17.2.2), and the last part summarizes the overall results on the tested public cyber datasets (17.2.3). HAL is noted to be robust against shared adversarial attacks targeting the VGG16 model and is scalable to different malware families and various cyber datasets due to the design of its two learning strategies. The code for reproducing HAL and experiments is available on GitHub.

### 1. Performance Comparison

The normalization items of each security strategy have been adjusted to achieve uniformly distributed weights ranging from 0.05 to 0.15, as outlined in the table. Given this distribution, the performance comparison of each strategy has been conducted within the determined ranges, excluding experimental outcomes derived from distributed weight configurations. Notably, both the proposed hybrid adversarial machine learning approach and the decentralized generative adversarial networks (DGAN) modification exhibit significant sample ratio improvements

over existing benchmark approaches (75 on DGAN, 84 on New GA, and 59 on NSGA-II). However, it should be emphasized that the hybrid adversarial machine learning strategy maintains superiority regarding both the average sample ratio (improved from 40.6260 to 75.1439, with a relative improvement ratio of 85.00%) and the maximum sample ratio (achieved as high as 84). The remaining peers (DGAN, New GA, and NSGA-II) yield negligible sample ratio improvements, with relative improvement ratios of no more than 5.00% [4].

The exploration results of the hybrid adversarial machine learning strategy have highlighted its robustness compared to other security strategies when only a portion of the samples are compromised or under attacks. The hybrid adversarial machine learning approach has demonstrated the capacity to optimize the defensive and offensive model structures simultaneously while considering the influence of the defender's knowledge in the modeling space. The experimental results have further revealed that the local model modification on the defender's side experimentally works worse than that on the attacker's side.

### 2. Robustness Evaluation

The robustness of the proposed Hybrid Adversarial Machine Learning (HAL) framework is evaluated by assessing its resistance to adversarial perturbations under diverse threat models and attack intensities. Robustness evaluation focuses on the model's ability to maintain stable performance when subjected to both known and unseen adversarial samples, particularly in black-box and partial-knowledge attack scenarios. As highlighted in [17], adversarial robustness should not be limited to accuracy on clean data but must incorporate resilience against transferable attacks, distribution shifts, and adaptive adversaries that exploit model vulnerabilities beyond the training distribution.

Following the evaluation principles outlined in [19], robustness is measured through systematic stress testing using adversarial manipulated samples across multiple datasets and attack configurations. The proposed framework is analyzed under conditions where only a subset of samples is compromised, reflecting realistic cyber-attack scenarios. The results demonstrate that HAL maintains consistent detection performance and exhibits reduced degradation compared to baseline defense mechanisms. This confirms that integrating adversarial training with defensive distillation enhances both local and global robustness of the model. Overall, the robustness evaluation validates the effectiveness of the hybrid approach in mitigating adversarial risks and supports its applicability in real-world cybersecurity environments where evolving and unknown threats are prevalent.

## VIII. DISCUSSION AND IMPLICATIONS

The results obtained in this study provide important insights into the effectiveness of hybrid adversarial machine learning strategies for enhancing cybersecurity defenses. As emphasized in recent evaluations of vulnerabilities in machine learning-based security systems [13], standalone defense mechanisms often fail to offer sufficient protection

against adaptive and evolving adversarial attacks. The experimental findings of this work confirm that combining complementary defense techniques significantly improves robustness compared to isolated approaches.

From a theoretical standpoint, the proposed Hybrid Adversarial Machine Learning (HAL) framework demonstrates that integrating adversarial training and defensive distillation leads to improved generalization under adversarial conditions. This observation is consistent with recent surveys on adversarial attacks and defenses in malware classification [4], which highlight the limitations of single-layer defenses and advocate for multi-stage or hybrid security architectures. By jointly addressing evasion, poisoning, and partial-knowledge attacks, the proposed framework aligns with modern threat models encountered in real-world cybersecurity environments.

From a practical perspective, the implications of this work extend to the deployment of machine learning models in operational security systems such as intrusion detection systems (IDS) and Security Operations Centers (SOCs). The improved robustness observed in the hybrid approach suggests that such systems can better withstand adversarial manipulation while maintaining acceptable detection performance. However, this increased resilience comes at the cost of higher computational complexity and resource consumption, as also reported in existing malware defense studies [4]. These trade-offs must be carefully considered when deploying hybrid adversarial defenses in large-scale or resource-constrained environments.

Overall, the findings reinforce the necessity of continuous evaluation, hardening, and adaptation of machine learning-based cybersecurity solutions. As adversarial techniques continue to evolve, robust hybrid defense strategies such as HAL represent a promising direction for building resilient and trustworthy AI-driven security systems.

### 1. Theoretical Contributions

Significant theoretical contributions are made by the findings in this study. First, understanding of hybrid adversarial machine learning addressing the cybersecurity landscape evolving from the application of Large Language Models (LLMs) is expanded. These ML models are subject to several existing attacks and a few possible new attacks increasing the efforts on the defensive side [2]. Several avenues are opened for the advancement of the AI controlling all cyber defense practices. Second, extensive possibilities are opened on the deployment and improvement of cyber defense practices making use of hybrid adversarial ML techniques able to replace or enhance Tier-2, Tier-3, and even Tier-1 SOC practices [13]. Largely understudied and undeployed by cybersecurity practitioners, LLMs constitute a disrupter in the cybersecurity domain and a potential game changer in the ML cyber defense field.

There are several theoretical contributions made by the findings in this study. First, the understanding of hybrid adversarial ML handling the cybersecurity landscape evolving out of the application of LLMs is expanded. Concerning the ML models, these are held subject to several existent attacks and a few new possible attacks are

sketched, increasing the efforts on the defensive side. Concerning the cyber domain, several avenues are opened for the application and improvement of cyber defense practices making use of hybrid adversarial ML techniques, able to replace or enhance the Tier-2, Tier-3, and even Tier-1 SOC practices. Largely unexplored and undeployed amongst cybersecurity practitioners, the LLMs constitute a disrupter of the cybersecurity domain and a possible game changer in the ML cyber defense field.

### 2. Practical Implications for Cybersecurity

The Cybersecurity Threat Environment in the United States tackles the cybersecurity threat in the United States. It begins with a description of the evolving threat as more Internet of Things (IoT) devices are deployed and more Artificial Intelligence (AI)-driven systems are implemented. The recent Ukraine cyberattacks and the emergence of a new crime-as-a-service ecosystem are used as examples of headline events highlighting the severity of the cyberthreat. This description is followed by a breakdown of how the threat is analyzed in terms of critical areas of concern, threats to the United States, and concerns about current capabilities and authorities [13]. This section closes with a synopsis of how the proposed study provides valuable data and methodologies that can be used to better understand and combat the cybersecurity threat.

Five Tangible Applications are identified, which are representative of pressing cybersecurity challenges faced by many agencies and industries, whether it is automakers guarding against data breaches from their fleet of connected cars or a school district grappling with questionable web content appearing in its schools. These use cases are of high priority and have direct practical implications as it pertains to real-world applications of the methodology [2]. The aim is to provide a broad overview of these challenges and how the proposed study's methodology could be adapted and tailored to address them.

## IX. FUTURE RESEARCH DIRECTIONS

There are numerous open research problems that can be explored in this domain. Some short-term objectives are: • The multi-task transferability of adversarial examples will be investigated to understand whether there are tasks where adversarial examples exist across a variety of related models. • A test for the general transferability of black-box attacks on homogenous systems (i.e., different instantiations of the same model architecture) in terms of model hyperparameters and design choices will be developed. • The black-box effectiveness of adversarial examples will be further evaluated under controlled real-world settings beyond image classification. Medium- and long-term objectives are: • Surrogate model solutions will be developed to explore the transferability of adversarial attacks across different architectures of models. • A generative approach to the modeling of population transferability of adversarial attacks will be investigated. This will involve developing population-level representations for both images and adversarial perturbations and modeling how they interact with each other. • New methodologies and approaches for multi-task black-box attacks will be developed. Such attacks will

exploit commonalities in the models' architectures, hyperparameters, learning objectives, and training datasets. The Intensity-Based, Multi-hop Black-Box Attacks will be further investigated. In particular, attempts will be made to broaden such attacks beyond the scope of image classification (e.g., verify their effectiveness on object-detection models). Also, adversarial attack methodologies that register the needs of risk-sensitive industrial applications will be developed.

## X. CONCLUSION AND RECOMMENDATIONS

Mitigating cyber threats remains a major challenge, owing to innovative cyber-attack strategies and complex software systems. The deployment of Machine Learning (ML) systems has augmented these challenges as they prevail over traditional methods in recognizing novel and sophisticated cyber threats. Nevertheless, the vulnerabilities of the ML systems, particularly due to Adversarial Examples (AEs), may hinder their widespread adoption. Recent adversarial attack methodologies could find undetectable attacks to the state-of-the-art deep learning models, constituting an enigmatic threat to the current architecture of cyber-defense systems based on deep learning technology. This framework is focused on accommodating pioneering attempts at predicting the adequateness of newly crafted Adversarial Attacks with respect to different ML Models under the black-box setting, where neither knowledge of the target model architecture nor its training set is presumed [13]. The work underscores the significance of continual checking of the hardening ML models against their vulnerabilities in the light of the newly framed innovative adversarial attacks and the exploration of architecture-independent safeguards, not prevailing in the fabric of recent vigilance in this domain.

There is a pressing need for extreme caution in adopting ML models in risk-sensitive industrial applications as the entire architecture, learning, and conduct of the security mechanism on AI/ML engines are being directed and trained comprehensively toward the legitimate domain view. For such systems, the security arrangements must be fortuitously foiled, fortifying future scrutiny endeavors for apt hardening of their cognizance [1]. Technological strides are required to ensure plausible security and privacy, such that the gracefully positioned defense systems remain incapable of hindrances to even the most stellar adversary, masked AEs. Speculating unknown and/or unseen percepts or executing something radically dissimilar from training dissimilar to the sanctioned model might lay their groundwork in the anticipatory unbending ML domain and budding vigilance in anti-adversarial research.


## REFERENCES


- [1] A. Oseni, N. Moustafa, H. Janicke, P. Liu et al., "Security and Privacy for Artificial Intelligence: Opportunities and Challenges," 2021. [PDF]
- [2] P. Y. Chen and S. Liu, "Holistic Adversarial Robustness of Deep Learning Models," 2022. [PDF]
- [3] K. Aryal, M. Gupta, and M. Abdelsalam, "A Survey on Adversarial Attacks for Malware Analysis," 2021. [PDF]
- [4] M. Datta Sai Ponnuru, L. Amasala, T. Sree Bhimavarapu, and G. Chaitanya Garikipati, "A Malware Classification Survey on Adversarial Attacks and Defences," 2023. [PDF]
- [5] D. Rios Insua, R. Naveiro, V. Gallego, and J. Poulos, "Adversarial Machine Learning: Bayesian Perspectives," 2020. [PDF]
- [6] G. Zizzo, C. Hankin, S. Maffei, and K. Jones, "Adversarial machine learning beyond the image domain," 2019. [PDF]
- [7] S. M. Devine and N. D. Bastian, "Intelligent Systems Design for Malware Classification Under Adversarial Conditions," 2019. [PDF]
- [8] A. Siddiqi, "Adversarial Security Attacks and Perturbations on Machine Learning and Deep Learning Methods," 2019. [PDF]
- [9] S. Kotyan, "A reading survey on adversarial machine learning: Adversarial attacks and their understanding," 2023. [PDF]
- [10] A. Kurakin, I. Goodfellow, and S. Bengio, "Adversarial Machine Learning at Scale," 2016. [PDF]
- [11] R. Rauter, M. Nocker, F. Merkle, and P. Schötle, "On the Effect of Adversarial Training Against Invariance-based Adversarial Examples," 2023. [PDF]
- [12] T. Anastasiou, S. Karagiorgou, P. Petrou, D. Papamartzivanos et al., "Towards Robustifying Image Classifiers against the Perils of Adversarial Attacks on Artificial Intelligence Systems," 2022. ncbi.nlm.nih.gov
- [13] J. Harshith, M. Singh Gill, and M. Jothimani, "Evaluating the Vulnerabilities in ML systems in terms of adversarial attacks," 2023. [PDF]
- [14] O. Ibitoye, R. Abou-Khamis, M. el Shehaby, A. Matrawy et al., "The Threat of Adversarial Attacks on Machine Learning in Network Security - A Survey," 2019. [PDF]
- [15] N. Carlini and D. Wagner, "Towards Evaluating the Robustness of Neural Networks," 2016. [PDF]
- [16] N. Papernot, P. McDaniel, X. Wu, S. Jha et al., "Distillation as a Defense to Adversarial Perturbations against Deep Neural Networks," 2015. [PDF]
- [17] S. Henrique Silva and P. Najafirad, "Opportunities and Challenges in Deep Learning Adversarial Robustness: A Survey," 2020. [PDF]

[18] S. Alam, Y. Alam, S. Cui, and C. Akujuobi, "Data-Driven Network Analysis for Anomaly Traffic Detection," 2023. [ncbi.nlm.nih.gov](https://ncbi.nlm.nih.gov)


[19] K. Steverson, J. Mullin, and M. Ahiskali, "Adversarial Robustness for Machine Learning Cyber Defenses Using Log Data," 2020. [PDF]

# Hybrid IDS Using Signature-Based and Anomaly-Based Detection

Messaouda Boutassetta   
Department of Computer Science,  
Faculty of Science and Technology  
LIMA Laboratory  
Chadli Bendjedid University  
PB 73, El-Tarf 36000, Algeria  
[m.boutassetta@univ-eltarf.dz](mailto:m.boutassetta@univ-eltarf.dz)

Amina Makhoulf   
Department of Computer Science,  
Faculty of Science and Technology  
LIMA Laboratory  
Chadli Bendjedid University  
PB 73, El-Tarf 36000, Algeria  
[makhoulf-amina@univ-eltarf.dz](mailto:makhoulf-amina@univ-eltarf.dz)

Newfel Messaoudi  
Department of Computer Science,  
Faculty of Science and Technology  
LIMA Laboratory  
Chadli Bendjedid University  
PB 73, El-Tarf 36000, Algeria  
[ne.messaoudi@univ-eltarf.dz](mailto:ne.messaoudi@univ-eltarf.dz)

Abdelmadjid Benmachiche   
Department of Computer Science,  
Faculty of Science and Technology  
LIMA Laboratory  
Chadli Bendjedid University  
PB 73, El-Tarf 36000, Algeria  
[benmachiche-abdelmadjid@univ-eltarf.dz](mailto:benmachiche-abdelmadjid@univ-eltarf.dz)

Ines Boutabia   
Department of Computer Science,  
Faculty of Science and Technology  
LIMA Laboratory  
Chadli Bendjedid University  
PB 73, El-Tarf 36000, Algeria  
[i.boutabia@univ-eltarf.dz](mailto:i.boutabia@univ-eltarf.dz)

## Abstract :

*Due to the ever-changing nature of cyber threats, intrusion detection systems (IDS) are now necessary for network protections against threat incidents. Traditionally, IDSs are classified as an IDS based on either signatures or anomalies. Each type of IDS has its own limitations that can include difficulty in detecting new threats, and high rates of false positives. The contribution presented in this research is a hybrid type IDS that can minimize the limitations and boost the performance in detecting a wider range of threats that include signature and anomaly approaches. The hybrid IDS detected threats using the Snort engine for signature detection and deployed a lightweight anomaly detection model that was trained using datasets such as KDD and IoT-20. The results indicated a significantly enhanced detection rate, substantial decreases in the number of false positive for both attacks, and improved periods/threat mitigation to respond to attacks in real-time. We examined potential uses of this hybrid IDS in selected industry settings such as in financial sectors, and air traffic control, and social networking, and evaluated the effectiveness of the hybrid IDS in realistically used case scenarios. Future developments may be in constructing a cloud based hybrid IDS with active models updates, and advanced AI and analysis of threats to respond to persistence of new and more sophisticated cyber-attack threats.*

**Keywords:** Intrusion Detection System (IDS), Hybrid IDS, Signature-Based Detection, Anomaly-Based Detection, Machine Learning (ML), Cybersecurity, False Positives, Detection Accuracy, Real-Time Detection, Network Security.

## I. Introduction

Both the societal dependency on computers and networking as a means of exchanging, manipulating, and storing data and the consequential emerging likelihood for abuses intended to disrupt the services provided by the contemporary dependency on such technology have spurred interest toward the innovation of Intrusion Detection Systems (IDSs)-an attempt

to protect computer networks, recently even taking into account the technological shift incessantly occurring worldwide towards wireless packet-based transmission environments [1].

As a response of Byzantine interest in security both by outsiders and insiders, an IDS serves two major However, it must be noted that due to the very nature of its intention-exposing the ongoing intrusion and relatedly acting on behalf of the list of agents forming the intrusion desire-the own design of an IDS must comply with a number of counteracting constraints. IDSs might be integrated at each host node using software packages, and/or be associated with special-purpose hardware-analysis devices operating passively on the link layer (layer-2) mechanism of the network [2].

### 1.1. Background and Motivation

Intrusion detection systems detect unauthorized access or exploitation of computer systems through event logs. A great amount of effort to search better solutions, new ideas, and approaches are proposed through decades to facilitate the task of detecting cyber intrusions [3]. In the last couple of years, there are also numerous benchmarks and publicly available data sets that stimulated advance research in the field. However, with the continuously changing essence of the attacks and the systems itself, conducting an effective and efficient IDS remains an on-going challenge that needs continuous endeavor for new solutions and better optimization of the existing ones.

Intrusion detection is based on either a signature- or an anomaly-based detection method. Signature detection systems use predefined signatures, such as byte sequences, byte counts, and other detectable patterns based on past knowledge. It is general knowledge that signature-based systems easily miss new and unknown attacks, and therefore need constant update in signatures - a process both time consuming and expensive [2]. Conversely, anomaly-based systems can detect unknown attacks, and as the name suggests, tries to model the normal use of the system in order to capture events that deviate from

this norm. However, various schemes of model creation have high false positive rates due to random non malicious deviations from the learning model and the possibility of exploits that mimic the normal use of the system attacking it. Both types of approaches have an important marketing acceptance, extensive research interest, and proposals in academia and industry. Some systems propose to combine the two separate processes. This, however, leads to heavily increased complexity and has not widely been used.

### 1.2. Research Objective

The aim of this research is to generate a hybrid Intrusion Detection System (IDS) using both signature-based and anomaly-based detection. A hybrid technique, which provides the benefits of both approaches, will be proposed by combining a group of statistical approaches with a neural network using the ideas of a conditional architecture. Intrusion Detection can be defined as the capability to detect unauthorized use of computer systems. An IDS can be classified into two general types: misuse detection and anomaly detection [4]. These two general types provide two different approaches for detecting intrusions. The first approach has also been traditionally referred to as signature detection. Under the misuse approach, the IDS must maintain an up-to-date database of intrusion signatures and find those patterns in the network. Even though this approach is becoming increasingly popular, it has a major drawback: it cannot detect new, novel attacks and stoics attacks [1]. On the other hand, the anomaly approach is based on the assumption that normal activity and intrusive activity will exhibit different behaviors and that an IDS can be designed to learn normal behavior. If it is used on the network, any deviation from that behavior would be detected as suspicious activity.

Anomaly detection systems are also not enough to act as a standalone solution to secure the systems. The combination of anomaly and misuse approaches has recently received increased interest as a means to provide a more robust detection capability. The goal is to take advantage of the strengths of both approaches while minimizing their weaknesses. In a hybrid scheme, a number of underlying independent detectors are combined. Each detector operates on its own, making independent decisions and the final decision is taken using a high level approach based on the combination of the decisions made by the individual detectors. The hybrid techniques based on the combination of multiple detectors are also classified into four types. A disseminative approach imitates an agent-based structure where each detector has its own knowledge of the data and the decision made on that data.

The remainder of this paper is organized as follows. Section II reviews background and related work on intrusion detection systems. Section III presents the proposed hybrid IDS approach. Section IV describes the design, implementation, and system architecture. Section V presents the results and discussion. Section VI addresses the challenges and future research directions of Hybrid IDS, and Section VII concludes the paper with a summary of findings and recommendations for future work.

The following section reviews background and related work on intrusion detection systems to provide context for the proposed Hybrid IDS.

## II. Background and Related Work

### Intrusion Detection Systems (IDS)

Intrusion Detection System (IDS) is an essential component of security for modern information systems. IDS can analyze several data sources such as network packets, application logs, databases, kernel logs, server log files, and many others. An IDS detects any misuse of information in an information system or computer system. It analyzes the activities occurring in real-time within the information system and determines whether or not the activities are valid. Specific rules are used to co-relate the data and these rules differ from one IDS to another. There are mainly two types of IDS [5].

Signature-based systems refer to those systems that maintain a signature database. Any new packet going into the information system is matched with the stored signature patterns. If a match is found, it indicates the occurrence of misuse or an intrusion and it generates an alert. Signature-based systems are good in achieving transparency because they analyze the activities that are occurring directly within the information system. They can also detect the misuse pattern with high reliability because they match specific signatures. However, it is difficult to define what a signature is and signatures consume a lot of space which leads to problems of false alarms when new attacks occur. Anomaly systems detect intrusions by searching for abnormal system activity [6]. They create a profile of what normal activity is for that information system initially. This profile is then used to monitor the behavior of the activities that are occurring. If a significant deviation from this profile is observed, it indicates the occurrence of misuse. Anomaly systems are good in terms of achieving coverage as they are not limited to a specific misuse pattern. They can detect known, known modified, and new misuse patterns. Anomaly systems also have low false alarm rates, especially for very large databases.

#### 2.1. Types of IDS

Intrusion Detection Systems (IDSs) can be classified into different categories, based on the monitoring approach. The classification can be made based on either the monitored platform or the employed technique. Based on the monitored platform, the IDSs can be either Host-Based or Network-Based. Host-based intrusion detection systems are responsible for monitoring a single computer system, while Network-based intrusion detection systems, which are devices or software components deployed in a network, analyze the traffic generated by hosts and devices [5].

Another important classification takes into account the employed technique. According to that, the IDSs can be of two types: signature-based and anomaly-based. If the detection system is based on cross-checking monitored events with a database of known intrusion experiences, the IDS is defined as signature-based. This approach requires an extensive database of attacks and is incapable of detecting new forms of attacks or attacks that have been redesigned to bypass detection. On the other hand, if the detection system is based on learning the normal behavior of the system and reporting whether some anomalous events occur, the IDS is defined as anomaly-based. The advantage of this approach is that it can detect new attacks that have not been previously documented [7].

## 2.2. Signature-Based Detection

Signature-based detection is one of the primary methods employed in Intrusion Detection Systems (IDS). Here, patterns of known threats and attacks are stored in the form of 18 signatures in the IDS databases [8]. Each intrusion signature contains information regarding the intrusion, source/destination addresses, and protocols used. Whenever a packet arrives, its header and contents are dissected for matching previously defined signatures. False alerts are triggered when a packet matches an entry in the signature list. The signature detection has two approaches: a complete signature file, which requires a lot of memory and processing resources, and a periodic scanning strategy, in which only a few bits of signature are searched at a time [5]. The first approach is capable of detecting the flood of attacks but can miss some other attacks, while the second one can detect all types of attacks but with a few bits of signatures.

Signature-based detection is effective if all attacks are known and characterized. Attack signatures can change over time due to modifications in attack tools and hacker activity or network environment changes. New attacks can be initiated that do not have signatures. Changes in normal behavior can be caused by the introduction of new software, modifications in the system and network environment, or variations in network load and user activities.

## 2.3. Anomaly-Based Detection

[5]. When calculating the probability of an event occurring, the time is considered, and an alert is raised if an event is unlikely to have happened in a specific time. Viinikka et al. exploits time series techniques by aggregating individual alerts into an alert flow, and examining it as a whole. This has the benefit to perform a more precise multivariate analysis and to lower the false positive rate of alerts. Qingtao et al. proposed a system focused on detecting abrupt-change anomalies of the computing system. They used the Auto-Regressive (AR) process to model the data and performed a sequential hypothesis testing to determine the presence of an anomaly. Zhao et al. exploited techniques to mine frequent patterns in network traffic and applied time-decay factors to differentiate between newer and older patterns. When developing an AIDS, attention must be given to data seasonality. Reddy et al. proposed an algorithm to detect outliers in seasonality-affect time series data using a double pass of Gaussian Mixture Models (GMMs) [2]. Knowledge-based AIDS falls in the category of expert systems. These systems leverage a knowledge source which represents the legitimate traffic signature. Every event that differs from this profile is treated as an anomaly. Walkinshaw et al. applied FSMs to the whole network traffic, representing the activities of the system by states and transitions. The produced FSM represents the nominal behavior of the system, and any deviation is considered an attack.

## 2.4. Hybrid IDS Approach

The core of hybrid IDS is the idea of combining the two aforementioned approaches, using both signature-based and anomaly-based detection methods. On one side, a lower number of false positives and a better detection speed can be achieved with the first approach, whereas improved detection accuracy can be gained with the latter approach. In order to benefit from the advantages of both families of detection algorithms, a hybrid detection paradigm is considered here through the integration of their two families [5]. In this hybrid

system, traffic flows are initially examined using signature-based detection methods. If a flow matches any of the given signatures, it is marked as an alarm. Otherwise, the flow is forwarded to an anomaly-based detection system that aims to detect novel attacks whose patterns do not match the given signatures.

To avoid all network flows from being forwarded to the anomaly detectors, which is also a performance issue, a selection step is added. After a flow is checked with the first family of detectors, it is classified as ham, and only the flows classified as ham are fed into the second family of detectors. In this way, no action is taken for flows whose signatures match the first family of detectors, excluding obvious attacks, such as common port scans or denial-of-service attacks using well-known tools. This decision also reduces the computational cost of the anomaly-based detectors by avoiding the analysis of several flows that are expected to be ordinary, such as web browsing, email transferring, and so on. Note also that this selection step can be implemented using either static rules or machine learning algorithms. For example, specific source/destination pairs of ports or addresses could be selected.

### 2.4.1. Integration of Signature-Based and Anomaly-Based Detection

A Hybrid IDS has been developed that integrates both signature-based and anomaly-based detection. These two modes of detection use different surveillance strategies, and each has been separately researched and implemented in a number of systems. Significant differences exist in the detection mechanisms. Therefore, the integration of these two detection modes is non-trivial. It is necessary that the Hybrid IDS integrates multiple modes of detection, choosing additional integration strategies. This research provides an overview of the processes and approaches involved in the integration of signature-based and anomaly-based detection. A technical discussion of the respective detection approaches, their integration, and other related issues is presented [3].

A surveillance system, either human or automated, gathers information about some environment and processes that information to find interesting patterns. Patterns of interest can be called alerts, and the environment being observed is the subject of surveillance, such as a network. The information available to the surveillance unit is limited or unstructured, so it processes the data to help simplify the task. In a computer network, for example, a spectacular amount of data is generated every second in the form of packet headers and audit logs. This data cannot be processed by human operators, therefore it is processed by automated systems called surveillance or intrusion detection systems (IDS). These systems can detect interesting activities such as attacks, misuse, or failures, and either react accordingly or present interesting findings to human operators. The simplest system is nothing more than a vigilant security guard, sequentially monitoring a single surveillance camera. This approach is passive, since the guard has no means to adjust to changes in the monitored system. An improvement to this system would be to automatically pass the cameras between locations that need more attention [5].

### 2.4.2. Advantages of Hybrid IDS

Hybrid intrusion detection has some advantages over single intrusion detection. This section confirms that idea by presenting the advantages of hybrid intrusion detection

systems. With the rapid advances in networking technology and the development of diverse and sophisticated network services, networks-based attacks have increased enormously. Hackers use various tools for network attacks and efforts are made to cover their malicious activities, making attacks a very complex and challenging problem [5].

Some intrusion detection systems (IDS) detect attacks at the network layer. Network-based (NIDS) and host-based (HIDS) are the two types of intrusion detection systems. Anomaly-based and signature-based are two basic strategies used by intrusion detection systems. Data mining techniques and statistical, and machine learning are also used as alternative techniques. Hybrid intrusion detection systems have been developed to combine network based systems with the other IDS to take advantage of more than one technique [2].

### III. Methodology:

#### 3.1. Design and Implementation

The design and implementation of the Hybrid IDS utilize both Signature-Based Detection and Anomaly-Based Detection from the previously proposed architecture. The design focuses on how to analyze the network traffic and which procedures would identify the attacks on the network. The system architecture design includes input data for analysis with the proposed method architecture and end-user representation. Data processing consists of how data is analyzed, processed, modeled into an open-source platform such as Snort, and how the deployed devices deploy signatures or rules [9].

The Design and Implementation section describes the architectural design and the data processing of the Hybrid IDS. The architectural design of the Hybrid IDS contains the network that needs analysis, devices deployed to collect the information from the network, and how the end user interacts with the deployed system. The network that will be implemented contains a 16-node network, and on this network, both Signature-Based and Anomaly-Based Detection mechanisms have to be implemented in the open-source NIDS Snort.

#### 3.2. System Architecture

An Intrusion Detection System (IDS) is an essential component of any secure computer network. Although security policies, firewalls, and access control mechanisms can reduce the chances of a successful attack, they cannot completely avoid intrusions. Unfortunately, the number of computer security attacks is increasing. Thus, it is necessary to install IDSs as an additional layer of security to protect computer networks against intrusions. The primary goal of intrusion detection is to discover unauthorized use and abuse of computing resources. The incidents can range from the violation of an organization's policies to actions that threaten its security [9].

The computer's operating system normally logs such activity, which can then be analyzed by the system's administrator. This process can be done during system operation but is normally executed on a periodic basis. The time that elapses between the intrusion and its detection increases the possibility of critical damage to data and resources. Therefore, as the computer system becomes more complex and the network grows, detecting attacks exclusively through traditional safeguarding approaches becomes increasingly difficult. It is also common for a sophisticated intrusion to

exploit system weaknesses over a period of time, which makes it invisible to the protection mechanisms. A better approach is to analyze the network traffic, transactions, or system calls/etc., as they occur [1]. This is the approach behind intrusion detection and the research presented in this work.

#### 3.3. Data Collection and Preprocessing

The process of gathering and preparing data for analysis within a Hybrid IDS, an important part of building an effective Intrusion Detection System (IDS). This includes the tasks and methods used to collect and organize data to ensure its quality and relevance, making it easier for the IDS to identify potential intrusions. A Hybrid Intrusion Detection System (HIDS) is designed by combining signature-based and anomaly-based techniques. Initially, a signature-based IDS is built to utilize various datasets. The advantage of the signature-based system is that it can discover attacks instantaneously once the signature is matched, especially for those attacks that are already known [3]. Nevertheless, a number of legitimate connections are falsely detected as attack connections, lowering the overall performance of the system. So, to increase the efficiency of this approach, the anomaly-based detection is subsequently developed to work together with the signature-based method. The firewall logs (TCP connections permitted) created by the signature-based detection are processed and pipelined into the anomaly-based detection engine. The novel incoming connections are assessed to be either normal or abnormal compared to the profile created from the training dataset. The data collection requires gathering logs from Intrusion Detection Systems (IDS) residing in each sub-network. If required, the gathered logs are filtered to remove redundant and false-positive alarms. Finally, the cleaned logs are stored in a database where they are indexed based on selected features to establish a dataset prior for analysis [10].

### IV. Evaluation and Performance Analysis

The proposed scheme tested both Network-based Hybrid IDS and Host-based Hybrid IDS by considering important metrics to evaluate the performance of the proposed technique. The following metrics have been used to evaluate the capability of the proposed Hybrid Intrusion Detection System (IDS) in terms of False Positive Rate, True Positive Rate, Detection Rate, and Precision [10].

**False Positive Rate (FPR):** The False Positive Rate is the ratio of the number of incorrect normal instances and the total number of normal instances.

$$FPR = FP / (FP + TN)$$

**True Positive Rate (TPR):** The True Positive Rate is also called as Sensitivity or Recall. It is the ratio of correctly predicted attack instances to the total number of attack instances.

$$TPR = TP / (TP + FN)$$

**Detection Rate (DR):** The Detection Rate of a model gives the percentage of actual positive observations correctly identified as in the case, the detected intrusions.

$$DR = (TP) / (TP + FP)$$

**Precision:** The Precision of a model gives the percentage of positive predictions that were actually correct.

$$P = (TP) / (TP + FP)$$

A comparative analysis has been made to measure the efficacy of the Hybrid IDS technique in terms of False Positive Rate,

True Positive Rate, Detection Rate, and Precision against the existing models. The performance has been evaluated for Network-based Hybrid IDS and Host-based Hybrid IDS based on Search-Tree and PATT-FED-Tree. The data has been taken from IoT- 20 datasets.

#### 4.1. Metrics for Evaluation

In the context of background & related work, intrusion detection systems (IDS) are most commonly used to defend against malicious attacks to the network-based systems and resources. The alarming rate of growth of the attacks towards the networked systems has raised the urge for an effective IDS. Investigating the existing anomalies that caused a breach in the network security of a system and intensifying it as an attack detection can help in development or suggestion of a minor changes that can avoid the same security breach in the intended future. A minor, cautionary incident picked up can work at an early stage and the system can respond in order to try to block the resources that intend on causing harm. Some of the metrics that are used to evaluate the performance of the hybrid IDS can be broadly classified as: intrusion cost (IC), throughput (TP), accuracy (ACC), false positive (FP), false negative (FN), true positives (TPs), and true negatives (TN). These metrics are collected and analyzed through experiments done under the NS2 and KDD datasets [11]. The performance of the hybrid IDS is then calculated & compared taking the routing metrics before & after the hybrid IDS implementation into account [12].

#### 4.2. Comparison with Single Detection Methods

To evaluate the performance of Hybrid IDS, the capability of Single Detection Methods is compared with Hybrid IDS. In this investigation, WSN-VT data set is handled in a Hybrid intelligent approach combining Signature-Based and Anomaly Based Detection methods. To specify such improvements in the performance of Intrusion Detection Systems, several metrics are established, such as True Positive (TP), False Positive (FP), True Negative (TN), False Negative (FN), Detection Rate (DR), False Alarm Rate (FAR). The Detection Rates of each Single Detection Method along with the Hybrid IDS is graphically represented in Figure 4. From the evaluation, the best approach appears to be Hybrid IDS. In addition, FAR could be reduced in Hybrid IDS up to 10%, irrespective of the type of attack [1].

The following section presents the results obtained from the evaluation and discusses their implications in real-world scenarios.

### V. Results and discussion :

#### 5.1. Case Studies and Applications

A real-world hybrid intrusion detection system combining signature-based detection and anomaly-based detection was constructed to explore the applicability of different algorithms on datasets from different environments over a long period of time. The hybrid system showed the validity of different algorithm application combinations and the user-defined environments and needs of certain combinations were highlighted, showing the flexibility of combination application. Issues in the compositions of hybrid systems and the applicability of the IDs components were mentioned [3]. Two specific components were discussed: 1) a method of combining a model-based IDS and a clustering-based IDS where the model-based IDS is used to support the clustering-

based IDS by preprocessing the data collected for the clustering analysis and providing alerts of grouped traffic; and 2) a discussion of how to classify new traffic off-line for already classified or known attacks in a hybrid IDS incorporating a clustering-based IDS and a signature-based IDS.

Further highlighting the potential and importance of the work, the possibility of future enhancements of the components and the hybrid systems was mentioned. Research still needs to be done on the design, analysis, and applicability of hybrid infection detection systems that integrate different types of computational paradigms and machine learning techniques [5]. Despite the research done on the separate categories of systems and/or implementation of certain computational models on datasets for specific types of attacks, very few systems exist that present the possible architectures and best techniques to integrate the systems to balance and maximize the performance of the improved intrusion detection system.

##### 5.1.1. Real-World Implementations

There exist some Hybrid IDS platforms in functioning or under construction. One of these systems is the Hybrid Network Intrusion Detection System (HNIDS) using both anomaly and misuse detection systems. The anomaly detection IIDS is based on the HMM model and the misuse detection IIDS is based on Bro, which monitors TCP connections and decodes the payload of packets to text [9]. HNIDS can detect both the known and unknown attacks. It was designed to work in a heterogeneous environment of different operating systems. The second system is a distributed and hybrid intrusion detection platform (AHDSP). The misuse detection tool used in the AHDSP is Snort 1.8.x, and the anomaly detection tool is an EC based IIDS (ECIIDS). ECIIDS is based on an evolving classifier with a capability of increments and decrements of classes. Second, mechanisms for the coordination of the processes of computers in the intruder monitoring network are proposed for improving the mining processes of knowledge on the user interests and further modifications of the database concerning the knowledge of behavior of normal users are presented.

The third was designed a hybrid intrusion detection model that combines multipartite graph-based representation of audit trails with hidden markov model based intrusion detection. The proposed system is adaptive to the addition of new audit trails and extends the capabilities of client or server audit trails for hybrid intrusion detection models (as discussed in results). The model is able to represent and detect complex types of intrusions matching multi-layered multi-site attacks and does not require the availability of a complete representation of all attacks prior to intrusion detection.

##### 5.1.2. Use Cases

This section presents specific scenarios and instances where the Hybrid IDS approach has been deployed to address security challenges. It highlights the diverse applications and contexts in which this integrated approach proves beneficial.

1. Air Traffic Control Systems (ATCS) This domain entails monitoring, coordinating, and controlling air traffic. The Hybrid IDS approach was employed to safeguard ATCS against unauthorized access and service denial attacks. It was trained to differentiate between regular, anomalous, and malicious air traffic monitoring activities. After a training period using input Normalized Data (ND) and corresponding output Target Data (TD), the system assessed real-time cases

based on raw input (RI) data. The approach successfully mitigated security threats with no false negatives. Statistical output reported no false positives during a specified inspection period.

2. Banking Systems Banking services on the Internet, or e-banking, involves a bank or financial institution offering online services to customers. Types of attacks in this domain include phishing, denial-of-service, and Trojan viruses. The proposed Hybrid IDS approach was trained using ri data and TD corresponding to a set of evolved bank transactions. Raw data of newly performed transactions was then considered residual and input into the system. The system accurately detected unauthorized access attacks on bank systems and verified customer transactions by system output. Corresponding alert systems also notified bank managers of attempted illegal operations, preventing resource damages or misuse [3].

3. Social Networks (SoNet) Social networks monthly cover around 160 million violations, with scores between zero and 10 indicating security risk. The deception is measured by knock-on/silent network perturbations, where inbound edges mislead users from one node to another. The Hybrid IDS approach employed knowledge-based deceptive detection methods and was trained on benign static SoNet data.

These findings highlight the practical relevance of the proposed Hybrid IDS and set the stage for discussing future challenges and research directions.

## VI. Challenges and Future Directions

Hybrid Intrusion Detection Systems (IDSs) utilize a combination of existing signature-based systems and newly developed anomaly-based systems to provide protection against a broad range of attacks without the negative impact on performance. Despite their utility and wide applicability, Hybrid IDSs have a number of limitations and challenges that remain unaddressed and require further attention. Only a handful of papers have been published to date focusing specifically on hybrid architectures that cover both detection mechanisms comprehensively [3]. Research addressing the specific challenges of Hybrid IDSs themselves is scant and needs to be expanded. For instance, research done on Signature Detection Systems consequently affects the Anomaly Detection Systems and vice versa. Very few IDSs exploit the trade-off between the two classes of systems to develop better classifications that reduce the constraints in each. Also, few if any, Hybrid IDSs would demonstrate sufficient adaptability to new attack methods.

The challenge of varying characteristics of different networks also needs further study [5]. It is important to develop adaptive hybrid systems that might adjust their models according to the characteristics of the monitored network. Such systems should be able to detect novel attacks without human intervention. Further studies are also needed to examine the impact an increase in anomalous traffic has on a Hybrid IDS. Comprehensive future research should be conducted on combining the two classes of systems into a single entity that autonomously handles both detection tasks. Data mining techniques have also been proposed as an effective way of discovering new attacks from audit trail data and of forming a new family of attacks that has a specific common behavior. These techniques should be investigated to determine their

potential effectiveness as a new strategy to overcome the limitations of all current intrusion systems.

### 6.1. Limitations of Hybrid IDS

The analysis of the collected data brought to the detection requirements that could create the architecture for a hybrid intrusion detection environment. The proposed architectures have strengths and weaknesses with respect to different criteria. For alert blades, even with the downsizing of false acceptable alerts in an intrusion detection system, there are still some degree of acceptable false alerts. For that reason, the solution is the cooperation of intrusion detection systems on the basis of false alerts analysis. The deployment of different technologies of intrusion detection systems allows the matching of alerts that can complement each other and increase the precision of possible threats [9]. There would still be situations in which even a combination of different technologies would not be enough to avoid false alerts.

On the other hand, there are potential attacks that use specific mechanisms to explore the weaknesses of intrusion detection systems, such as white-collar attacks. These attacks are performed by well-informed agents, with knowledge of the organization they are targeting and the techniques of the deployed security mechanisms, using this information to devise an intrusion that remains unnoticed by the intrusion detection systems. Each intrusion detection system provides a partial view of the entire monitored environment. This is another area where the deployment of multiple intrusion detection systems improves security. By complementing each other, they can provide a complete view of the environment. The observed strengths and weaknesses of a given architecture can determine the best possibilities for cooperation with other deployed architectures. In this sense, analysis of the possibility of communication and cooperation of the intrusion detection systems deployed by the same organization is necessary [1].

### 6.2. Research Opportunities

The Research Opportunities section sheds light on the potential areas for future research and development within the domain of Hybrid IDS. The focus is specifically on addressing metrics such as the efficiency, performance, adaptability, and reduction of complexity; exploring the unexplored avenues of IDS which are the combinations of recent trending techniques such as machine learning and artificial intelligence or rules-based data filtering in Hybrid IDS secures world-wide applications; conducting comprehensive analysis and testing of different datasets on Hybrid IDS with comparative evaluation; and finally exploring the Hybrid IDS on Cloud platforms and OS for better services, which are currently in the burgeoning growth stage. All of the above-stated unexplored opportunities could lead to the advancement of Hybrid IDS in terms of research and commercial viability which sufficient vendors and users equally [5]. For ensuring the robustness and future performance of Hybrid IDS systems, further ahead designing possible scenarios for the testing of efficiency and robustness of Hybrid IDS under attacks such as DDoS, Directory, and other ill behaviors. Moreover, for avoiding future loss of data, academics and industry researchers must analyze the vulnerability of rare attacks on the Hybrid IDS which were not used for the testing of performance of Hybrid IDS modeled from the credibility of datasets [9].

Addressing these challenges and opportunities paves the way for further improvements and the broader applicability of Hybrid IDS.

## VII. Conclusion :

The study and analysis of hybrid Intrusion Detection Systems (IDSs) using both signature-based detection and anomaly-based detection as an additional second level of detection have led to some findings, conclusions, and recommendations. The use of the K-Means clustering algorithm, coupled with supervised learning using decision tree techniques, results in a high-performance IDS capable of filtering high transmission rates of packets through a well-designed preprocessing architecture. A performance cross-validation of the IDSs is seamlessly integrated into the overall IDS architecture during the training process of the decision-tree-based models. Upon completion of the architecture and training of the models, hybrid IDSs are employed during the testing process, resulting in the detection of misuse and novel types of intrusions with a high degree of success, indicated by low misclassification rates and false-negatives object counts.

The performance of the hybrid IDS is further improved through the filtering of false alarms using the ensemble-learning approach. The counting of these false alarms is independently recorded for each of the models deployed in the ML-Hybrid IDS and concurrently fed back into the retraining process of the models. The retraining process is incremental and continuously runs along with the operational IDS and must be carefully managed not to incur any degradation in the overall IDS performance. The trained models are also evaluated in a continuous manner and must be able to account for any performance deterioration due to changes in the environment or the occurrence of new types of attacks. Should any of these performance thresholds be surpassed, the models must be immediately retrained from scratch. This overall scheme is an overview of the ongoing research efforts. Future work includes, among others, the selection of a priori probability for each of the classes upon their first appearance within the environment of the IDS and the experimental evaluation of the effect of this probability on the run-time performance of the IDS, as well as the development of a new hybrid second level of detection using ensemble classifiers instead of a single decision tree for the anomaly detection level.

### 7.1. Summary of Findings

Summary of Findings subsection encapsulates the key discoveries and results obtained from the research on Hybrid IDS using signature-based and anomaly-based detection. It offers a concise overview of the significant findings presented in the document.

A network hosts a web service that can be corrupted: that is to say, an intruder can initiate a series of operations/commands on the legitimate network merchant server such that the affected user accounts are tricked into approving unauthorized transactions, all while aiming to evade the scrutiny of the log analysis processes typically performed by the network intrusion detection system [1]. In an effort to thwart such activities, the network intrusion detection system models the legitimate behavior of the web service, the database, as well as the logging policies, and tasks the web service with a complementary logging mechanism able to record all executed

database transactions [9]. Such logged operations can be modeled as a timeline of data record modifications: additions, deletions, as well as the alteration of data values. This sequence of modifications is assembled into a description of the event activity record stream. Detection of potentially misleading operations requires the definition of the unwarranted execution of events/commands and also the grouping of event activity records comprising a relevant time window. That's where particular constraints about alterations of data record modification types are applied. By examining the database, user accounts are mutually linked; that is to say, the web service is able to disclose all transactions performed by a given user account. However, a small amount of legitimate wiring modifications can be performed on behalf of unrelated users. Each authenticated transaction is provided with a timestamp, and every legitimate grouping of modifications should abide by the temporal ordering of these timestamps.

### 7.2. Recommendations for Future Research

Hybrid IDS combines signature-based and anomaly-based detection mechanisms. Although this work makes an effort to come up with a Hybrid IDS that uses multi-technique detection methods, the use of only two datasets to assess the hybrid intrusion detection system is relaxation. As there are many datasets available presently in the field of intrusion detection, it falls short of consideration of several well-known datasets in the assessment of its hybrid detection model. However, future work can consider some of the well-known and inadequate datasets like UNSW-NB15 dataset and ISCX-IDS2012 datasets on top of the two chosen datasets in this work. In consideration of this, it is also far-fetched to claim the hybrid detection model is dataset oblivious [9]. Since the score based method is used to combine the output of multiple models, it along with the models can be varied for a suitable combination. Trained models can be saved and modified for future training, transfer learning and a parcel of best trained models can also be combined to provide a more robust detection model.

Another challenge that still remains unattended is a stateful analysis of the network traffic. This work conducted only a packet based analysis of the sniffed packets in the network. Future work can consider inspecting the packets for a time window to detect attacks on a session based environment. By doing this, denial of service attacks can be detected aside from sniffing the individual packets. Presently, many attacks like DDos are being launched and its detection is very hard with only the packet based analysis. By maintaining the state of the sessions, the detection of such attacks can also be performed which is a serious shortcoming in this work on top of the future prospects already mentioned [1].

In conclusion, the proposed Hybrid IDS significantly enhances network security and provides a strong foundation for future research in intrusion detection.

### References:

- [1] M.H. Bhuyan, D.K. Bhattacharyya, and J.K. Kalita, "Survey on incremental approaches for network anomaly detection," Tech. Rep., 2012.
- [2] Z. Zohrevand and U. Glässer, "Should I raise the red flag? A comprehensive survey of anomaly scoring methods toward mitigating false alarms," Tech. Rep., 2019.

- [3] G. Hendry, "Applicability of clustering to cyber intrusion detection," Tech. Rep., 2007.
- [4] M.H. Kamarudin, C. Maple, T. Watson, and H. Sofian, "Packet header intrusion detection with binary logistic regression approach in detecting R2L and U2R attacks," in Proc. Int. Conf. on Information Security, 2016, pp. 45-52.
- [5] P. Spadaccino and F. Cuomo, "Intrusion detection systems for IoT: opportunities and challenges offered by edge computing and machine learning," J. Inf. Secur. Appl., vol. 54, pp. 102-117, 2020.
- [6] Y. Sani, A. Mohamedou, K.A. Ali, and A. Farjamfar, "An overview of neural networks use in anomaly intrusion detection systems," in Proc. Int. Conf. on Cybersecurity, 2009, pp. 67-74.
- [7] J. Sen and S. Mehtab, "Machine learning applications in misuse and anomaly detection," Tech. Rep., 2020.
- [8] S. Reddy and V. S., "Signature searching concerning association assortment of files," Tech. Rep., 2015.
- [9] P.M. de Freitas Alves, "Analyzing audit trails in a distributed and hybrid intrusion detection platform," Tech. Rep., 2016.
- [10] M.A. Talukder, K.F. Hasan, M. Manowarul Islam, and M.A. Uddin, "A dependable hybrid machine learning model for network intrusion detection," in Proc. Int. Conf. on Computer Networks, 2022, pp. 120-128.
- [11] N. Munaiah, A. Meneely, R. Wilson, and B. Short, "Are intrusion detection studies evaluated consistently? A systematic literature review," J. Softw. Eng. Res. Dev., vol. 4, no. 2, pp. 1-14, 2016.
- [12] V.N. Tiwari, K. Patidar, and S.S. Rathore, "A comprehensive survey of intrusion detection systems," Tech. Rep., 2016.

# IoT Security Using Blockchain and AI

Djabar Abbas  
Departement of Computer Science  
LIMA Laboratory  
University of Chadli Bendjedid  
El Tarf, PB 73, 36000, Algeria  
dj.abbas@univ-eltarf.dz

Abdelmadjid Benmachiche  
Departement of Computer Science  
LIMA Laboratory  
University of Chadli Bendjedid  
El Tarf, PB 73, 36000, Algeria  
benmachiche-abdelmadjid@univ-eltarf.dz

Makhlouf Derdour  
Departement of Computer Science  
LLAOA Laboratory  
University of Oum El Bouaghi  
Oum El Bouaghi, 4000, Algeria  
derdour.makhlouf@univ-oeb.dz

**Abstract**—This paper explores the integration of blockchain technology and artificial intelligence (AI) to enhance security in Internet of Things (IoT) ecosystems. It addresses critical IoT vulnerabilities—such as data tampering, sybil attacks, and DDoS—by leveraging blockchain’s decentralization, immutability, and smart contracts, alongside AI’s real-time anomaly detection and adaptive threat response. The hybrid approach ensures data integrity, privacy, and trust landscapes, real-world case studies (e.g., secure biogas plant monitoring), and implementation challenges like scalability and resource constraints. It concludes with future directions, including federated learning, lightweight consensus protocols, and incentive-based trust models, advocating for interdisciplinary research to optimize this synergy for scalable, secure IoT deployments.

## I. INTRODUCTION

The Internet of Things (IoT) refers to the interconnection of physical devices and objects that are embedded with sensors, software, and other technologies, enabling them to collect and exchange data over the internet. IoT has gained significant attention in recent years due to the proliferation of affordable computing devices, advanced communication technologies, miniaturization, and embedded systems, which have made everyday objects smart and connected to the internet. This technology is enabling a wide range of applications across various domains, including home automation, healthcare, smart cities, smart energy, and agriculture.

However, the widespread adoption of IoT has also raised significant concerns regarding the security and privacy of personal data. As the IoT ecosystem expands, the number of devices connected to the internet is expected to increase exponentially, presenting significant challenges to address potential attacks and adversaries targeting these devices, networks, and the sensitive data they produce. Several measures can be taken to secure both the device and its data transmission to the cloud, to mitigate the risk of attacks on both the device and its data transmission to the cloud. In particular, blockchain technology has emerged as a potential solution to secure IoT devices. Within the IoT Frameworks initially proposed as a

decentralized approach, PILOT is the construction of a distributed ledger containing a list of records obfuscated and chained by a set of smart contracts, executed in a peer-to-peer network of nodes that governs the state of the IoT devices [1]. Compared to the historic centralized frameworks, with both a single authority and point of failure, the integration of the blockchain can dramatically reduce the impact of certain attacks to IoT devices, networks and data. Nevertheless, the designs of the architecture should be carefully considered in order to mitigate the risks of more recent threats, such as Sybil attacks. These concerns are even greater in the case of public ledgers, where transactions are not restricted to a single group, and thus, any node would attempt to impersonate or manipulate the network and its devices. Also, in the scope of the T framework, the number of transactions is expected to increase widely, presenting new challenges regarding performance and cost [2].

### A. Background and Significance

The Internet of Things is the next step in the evolution of the Internet and aims to interconnect common physical items to provide people with better lives via automated services using low-cost sensors, AI algorithms, and real-time data [1]. The interaction between, and cooperation of, these smart physical objects will generate data-centric models, in which smart contracts will run on the blockchain for data processing and application development, depending on the performance of the devices and their storage capabilities. Blockchain is an utterly decentralized technology that has the capability to facilitate value exchange over the Internet. Privacy and security are essential aspects to consider since myriad general blockchain applications, such as Bitcoin, Ethereum, and Alibaba, are currently being developed. Several external attacks have been discovered on private data and operation execution in the existing solutions, such as Sybil attacks, long-range attacks, and Distributed Denial of Service attacks. In order

to integrate safety and transparency requirements into IoT systems, a blockchain model is proposed here for the multidisciplinary collaboration of physical items. Using a hierarchical structure containing the IoT, fog nodes, and cloud nodes, which presents a wide range of study options, the model helps overcome challenges such as single points of failure, heavy workloads at the centralized data store levels, a lack of transparency, and poor scalability [2].

On a permissionless blockchain, such as Bitcoin, the presence of open information on transactions has led to a new kind of attack based on external data analysis. Methods for clustering user addresses revealed a deep behavior of these currencies' users, which goes against the anonymity they claim to guarantee. Specifically, the mathematical evaluation of the models and algorithms until now proposed helps understand what is achievable in terms of safety and which human behaviors encourage illicit activities. In this model, the violent emergence of geographical zones is non-growth wherever transactions are sufficiently high. In addition, notions of cryptography that employ smart contracts in an obsolete manner can happen and lead to the awakening of aberrant behaviors. Moreover, the interdependence between cryptocurrencies and social networks becomes significant on large scales: a model for the prices' relationships is proposed here, so that sudden price rises on any of them trigger a social response. In all, these findings may help prevent any use of these currencies as a mechanism for illicit activities, despite their attractive features.

#### *B. Research Objectives and Scope*

There are huge numbers of connected low-power IoT devices; hence securing them becomes a hard issue using traditional mechanisms. This dissertation proposes the use of novel technology blockchain with artificial intelligence (AI) for the IoT security improvement. Blockchain technology can be used to secure millions of IoT devices and protect them against cyberattacks. Reviewing the existing studies, it is identified that there is a lack of research regarding the IoT security improvement with AI-based blockchain technology. Hence, there is a need for research in this area.

The main objective of this dissertation is to identify the works done in the area of the application of AI-

backed blockchain mechanism security improvement in the IoT setup. To accomplish this several sub-objectives are defined, which are: (1) To study about the IoT and its security issues, (2) To research on the background of blockchain, AI and its application in cyberspace, in order to secure the devices and networks, (3) To do a thorough literature review on current IoT security approaches using blockchain and AI, (4) To identify the gap and come up with the future research direction for these technologies implementation with IoT to improve its security [2].

## II. FUNDAMENTALS OF IoT SECURITY

With today's advancements in technology and the Internet of Things (IoT), the unique identification of connecting objects and their access to the internet has come up to a higher level and a large number of devices are connected to the internet, empowering the "smart" world technology [1]. Security concerns regarding the devices and data breach are part of these technologies. IoT systems suffered from various issues such as data integrity. These device systems are often integrated with operations that can have a real effect on the physical world and strong resilience properties using atomicity, validity, isolation, and durability (AVID) lists are non-trivial in standard programming languages and their lack of transparency in working with integrity could lead to due clarifications of malintent or unintentional actions that would yield questioning its provision. In the sore aspects of the deciding actors, there would converge an irreplaceable trust in the abstract.

Conventional IoT models provide solutions to these concerns considering a coding approach, but they are vulnerable to attacks such as denial of service, replay, and man-in-the-middle which renders the non-centralized point idea useless [2]. Therefore, the lack of clear direct accountability or interventions for the actors executing the actions/decisions runs ill-governed this functional necessity of technology, product security evaluation does not tend to question on-software breaches under the apply rule of any absolution debate. Blockchain and IoT together can provide a decentralized trust-oriented model that serves the vulnerability and safety concerns of a wide area such as banking, health, organizational development, and overall building smart cities. The

difficulties and practical aspects of implementing this convergence benefit in combating external assaults or incorrect transactions would raise the development potential to further aspects such as full independence and ensure safety.

#### A. Key Concepts and Definitions

Key concepts and essential definitions related to the IoT security are elucidated in this subsection, laying the groundwork for grasping the intricacies of safeguarding interconnected devices and systems. Most of the terms described here are directly relevant to the implementation of security for the IoT domain using blockchain and AI, and they will be more extensively discussed in the following sections as necessary.

The Internet of Things (IoT) refers to a worldwide network of interconnected physical objects or “things” that accumulate and exchange data on the Internet. These IoT objects, equipped with sensors, software, and other technologies, have as a common trait that they can effectively monitor and manage these technologies in order to collect and exchange data [1]. The generational evolution of IoT systems is subdivided into 3 (three) components (1) Smart Devices accumulates data from the environment through sensors and are located, identifying them by an IP address, (2) Smart Applications consumes data, processes information, and takes the necessary actions, and (3) Networks transports data and information between smart devices and smart applications. An extensive variety of IoT devices feed smart applications with data, including IP cameras, smart doors, and other smart home automation devices, as well as commercially available smart traffic control lights, vehicle parking lots, smart power meters for cities, and automatic irrigation devices for agriculture. In the agricultural sector, low-power WSN devices can monitor environmental variables such as temperature, soil moisture, air humidity, and CO<sub>2</sub> levels.

#### B. Challenges in IoT Security

The Internet of Things (IoT) has numerous advantages that stem from its ubiquity and potential for mass adoption, making it a new frontier for developing next-generation devices, data processing architectures, algorithms, automated services, etc. The convergence of IoT and cloud computing gives rise to a multitude of advantages and cutting-edge capabilities such as

ubiquitous computing and always-on demands. IoT enables access to a vast amount of data, which, when integrated with AI techniques like deep expansion, is expected to lead to remarkable developments in various fields, such as intelligent transportation, smart healthcare, industrial automation, precision agriculture, and energy grids. This can translate into valuable business insights, better resource management, lower operating costs, and new and highly personalized services [1].

However, with the emergence of the IoT paradigm, the gigantic ecosystem of smart interconnected devices comes with notable abilities, challenges, and vulnerabilities. More specifically, this new world entails several challenges and obstacles, such as the diversity of devices, networks, and services; unreliable and heterogeneous communication channels; new types of edge devices; high mobility; stringent real-time restrictions; limited resources, battery, bandwidth, and Computational power; and a greatly expanded attack surface [2]. Restrictive system and end-user generated potential breaches for compromising the customer experience. Unauthorized or dishonest devices misrepresenting measurements or computations, unprotected devices experiencing brute-force attacks, compromised devices generating dishonest reports or even false data incorporation, privacy breaches, and leaking sensitive information are deficiencies to name a few. The aforementioned challenges, difficulties, and vulnerabilities in the environment of the IoT are driving motivations and paving pathways for researchers and academics to tackle how to guarantee a secure and resilient IoT ecosystem.

### III. BLOCKCHAIN TECHNOLOGY IN IOT SECURITY

As the growing number of connected machines and devices have changed the way humans perform tasks both in daily lives and workplaces, the development and deployment of smart cities with an expansive Internet of Things (IoT) have received significant momentum from industries and academia. Numerous industries have already deployed IoT-based applications in smart home, healthcare, agriculture, and environment. On the other hand, technological giants such as IBM, Microsoft, Oracle, etc., are providing IoT platforms to assist industries in developing and deploying IoT ecosystems. Despite the assurance of safety and security, public concerns regarding

the trustworthiness of the entire IoT ecosystem have come to light, bringing the vision of IoT to the verge of disruption [1]. Because it is impossible to be monitored and trusted, connected machines are becoming the victim of manipulation, hacking, spoofing, DoS/DDoS attacks, etc. To establish transparency, accountability, and responsibility over the IoT ecosystem in a decentralized way, Blockchain technologies have been considered to be integrated with IoT ecosystem.

Blockchain is a decentralized, distributed ledger that keeps a list of records (or blocks) in a way that ensures data consistency, tamper-proofness, immutability, non-repudiation, confidentiality, availability, audit capability, accountability, and fairness. Various aspects and applications of Blockchain technology (with smart contracts) have been discussed, including, but not limited to, design, development, and deployment of decentralized marketplace, supply chain, cryptocurrency, etc. Apart from cryptocurrencies, Blockchain technologies have been considered to be implemented in a variety of domains, such as finance, healthcare, insurance, energy management, transportation, high-performance computing, and many more. Blockchain technologies have already been successfully implemented in many industrial IoT use cases, such as digital asset trading, data sharing in transport, event ticketing, and vehicle insurance [3].

#### *A. Overview of Blockchain*

Blockchain is a decentralized technology that implements a secure distributed ledger. It stores the transaction record across several computers, known as nodes. Whenever a transaction occurs, it is encrypted and stored in a block, which is later appended to the previous block and forms the chain of blocks. For better transparency, this chain is shared across multiple nodes and is replicated in multiple copies. Thus, when a hacker wants to alter a record in the transaction database, he/she has to alter the same record in every copy of the chain available in every node. The blockchain implementation involves three core technologies: cryptographic keys, a peer-to-peer network, and protocols for computing and storing transactions [1]. The blockchain architecture is illustrated in Figure 1.1. This section describes how blockchain works, its properties, and the types of blockchain.

The core working principle of a blockchain involves first the creation of a transaction by a participant. This transaction will later be encrypted using cryptography. The encrypted transaction is then sent to a network of nodes. Each node independently verifies the transaction using consensus standards. If the transaction is valid, it is bundled with other valid transactions to form a block. Each block is encrypted itself along with the nonce. The nonce is a random number generated for the block, which helps in forming a hash. The hash of the block, with the hash of the previous block is later stored in it. This creates a link between the blocks and protects against tampering since only a small change in the block will require redoing all the work with all subsequent blocks. The newly created block is then broadcasted to the network. Other nodes that receive the new block verify it and add it to the chain, and more copy of the updated chain is created. The security of the blockchain is guaranteed by its properties. In order to alter a transaction within a block, the hacker needs to first find the correct nonce which is a very computationally expensive task. So, in order to alter a block, it is necessary to redo the work for that block and do the work for every following block which makes the alteration nearly impossible. The properties of blockchain that make it secure and distinguishable are: immutability, transparency, decentralization, and batch process [3]. There are different types of blockchain that can be used based on the requirements. These types include permissionless blockchain that allows anyone to read or write, and permissioned blockchain that needs permission to write or read in the chain. There are also open blockchain systems and private blockchain systems.

#### *B. Applications of Blockchain in IoT Security*

The introduction of various “Things” controlled via Internet raises the question of security. Today’s IoT is not as secure as one would like. Aid of Blockchain and AI is being actively solicited to act as a panacea for not only security but also privacy and dealing with big data. Understanding the applicability and limitations of Blockchain and AI for IoT is now of utmost importance. Byzantium’s General and the onboarding problem highlight on properties that a Node should have before being part of the Blockchain. Ether uses proof of work mining and the concept of gas in contracts to limit the rather uncontrolled spending of computational

power. More controllers regulated currently by a vendor could profit significantly by moving to an Open IoT via Blockchain [1]. This way they would become responsible for the data they process and hence own its monetizing potential.

Another approach is multi-chains. Different things with various policies could then communicate in one network ruled by multiple chains. Industry 4.0 also seeks the establishment of a secure and efficient manufacturing ecology with a decentralized Industrial Internet of Things (IIoT) [3]. In security systems, Blockchain provides shared distributed ledger ensuring secure tamper-proof transactions between various users in the network without involving third parties. Potential applications of Blockchain in IoT are considerably vast, covering industries like agriculture, Insurance, Automotive, Healthcare, Manufacturing, Supply Chain Management. Together IoT and Blockchain help in real-time monitoring of resources.

#### IV. ARTIFICIAL INTELLIGENCE IN IoT SECURITY

Artificial intelligence (AI) has evolved tremendously and is now a part of our daily lives, making technology more human-like. With the advancement of AI technology, there are many applications through which machines and devices can mimic human intelligence. Industries use AI technology to control smart functional devices at a low cost, consuming low energy for various assistant-based devices like smart TVs, Alexa, and Google Home to control smart home appliances. AI technologies, namely machine learning (ML) and deep learning (DL), are emerging technologies for the industrial internet of things (IIoT) and provide various solutions [2]. The proliferation of IoT technologies results in controlling devices through the internet which connects everything around us. In this smart ecosystem, vulnerabilities and attacks have emerged, making security and privacy a major concern.

Hence, to reduce vulnerability and ensure safe communication, an AI technology-based security module is enforced to create a protected environment for communication. Common attacks in IoT are denial of service, malware attack, spoofing attack, MitM attack, eavesdropping, and various internal attacks. Several security techniques are in place to protect smart devices and IoT networks, but they use conventional approaches which have major drawbacks and are

not suitable in an IoT environment [1]. AI is a new and advanced technology expected to transform the industrial horizon and bring a new era. AI in IoT makes analytics smart and builds an intelligent system that ensures operators take the right decisions at the right time.

##### A. Introduction to AI

Artificial Intelligence or AI is a branch of computer science that mimics human intelligence through machines. This technology is based on 3 key concepts which govern the basic principle behind AI's functioning. They are learning, reasoning and self-correction. The learning concept is based on gaining information and knowledge regarding any object or technology. The reasoning concept is based on interpreting the situation and making decisions from the information gained about the object and technology. The self-correction concept is a way of correcting the mistakes that are made during the first two processes. There can be possible mistakes in gaining knowledge or interpreting them. AI is focused on discovering the mistakes that are made and correcting them [1]. This technology can be helpful in various fields like engineering, astronomy, medicine, etc. One of AI's significant importance is in the field of security. It can be used in surveillance systems, anomaly detection and risk assessment.

Various devices such as sensors, cameras, and RFID can be integrated into IoT to Monitor and Control real-world processes. IoT is the interconnection of various smart devices and communication technologies that collect and transfer data over the network [2]. This smart environment can be beneficial for smart cities by increasing their precision and efficiency. IoT applications are vulnerable to various security attacks from attackers. To defend against the attackers, AI algorithms can be implemented in the IoT system for anomaly detection and threat identification in the collected data. These detected anomalies can be used in response to recovery actions in the system, message filtering, and block creation in the Blockchain structure.

##### B. Applications of AI in IoT Security

Intrusion Detection System (IDS) Based on Machine Learning: The Intelligent Analysis of Malicious Attack Scenarios. With the development of Artificial Intelligence (AI), Industry 4.0 emerges with the convergence of

emerging technologies such as IoT, Cloud computing, big data, 5G, and AI. AI in Industrial Internet of Things (IIoT) aims to the automated intelligent decision-making devices, which are completely integrated with each other for serving various industrial processes. Recently, data breaches happened in the IIoT environments involving the production of counterfeit products and compromising the safety of human operators in smart factories. Without human intervention, the trustable and intelligent devices are required to carefully observe the illicit events and intelligently analyze it through the understanding of the malicious intentions behind the events. This research work provides the intelligent analysis of attack scenarios using AI for recognizing the malicious intentions of inducing attacks in the IIoT environment.

Artificial Intelligence (AI) lightweight blockchain security model for security and privacy in IIoT systems: This paper intends to develop an Artificial Intelligence (AI) lightweight blockchain model for security and privacy in Industrial IoT (IIoT) systems. IIoT data security and privacy threats are analyzed, and their requirements are presented. An AI lightweight intelligent blockchain security model is designed. A privacy-preserving AI model is proposed for the classification of attacks. Blockchain is employed for privacy preservation. The model employs Convolution Neural Networks (CNN) for IoT (IoT) data embedding and prediction. The toxicity of embedding parameters is preserved. The blockchain uses hash chains for making sure the integrity and availability of data. Transactions of the AI model on the blockchain are verified under the Consensus mechanism. Fuzzy logic-based trust is implemented for ensuring the correct behaviors of the IoT and IIoT devices. The replay attacks are identified by checking timestamps. The proposed scheme's efficiency and robustness concerning security attacks are proven [2].

#### V. HYBRID APPROACH: BLOCKCHAIN AND AI INTEGRATION

Artificial Intelligence (AI) and Blockchain technologies have drawn attention in enhancing security mechanisms for Internet of Things (IoT) ecosystems. Blockchain technology, based on distributed and decentralized nature, provides a tamper-proof distributed ledger for securely storing data [3]. Smart contracts are the crux of blockchains, automating regulations through

programmable rules [2]. Blockchain technology can help with IoT authentication and firmware updates and provide end-to-end communication privacy and integrity. Blockchain, however, cannot guarantee or verify the honesty of the data shared among participants, as sensor data may be tampered with before being recorded in the Blockchain.

To address such scenarios, advanced AI models can be deployed alongside Blockchains. AI models trained to simulate the characteristics of a process are capable of identifying anomalies and/or intrusions in the network. Anomaly detection can be used with blockchain-based authentication and access control to prevent intrusions in the first place. Although many studies focused on the individual advancement of AI and Blockchain technologies to secure IoT environments, there exists an immense cross-disciplinary opportunity to exploit the synergistic integration of both technologies.

##### A. Benefits of Hybrid Approach

Combining blockchain and AI has advantages and strengths that neither technology can provide alone. Together, a new set of benefits and capabilities emerge. Adopting a hybrid approach, particularly when dealing with sensitive and critical applications, provides a compelling case for enhanced IoT ecosystem security. Blockchain can ensure data integrity for the data inputted and outputted from AI models in the training and inference stages. This addresses the trust issues of organizations that rely on cloud-based AI models, ensuring that the models operate on the data as intended without malicious interference. AI can aid in the detection of suspicious behaviors in smart contract execution and the prediction of future trustworthiness scores. This tackles the challenge of fraud concerns in blockchain networks, allowing for the identification of deception quickly and efficiently. On the other hand, only with a hybrid approach does the combination of blockchain and AI technologies anchor on a unique structure of features, particularly attributes, to securely mine association rule data from device environments in a federated manner [3].

Many potential advantages arise from the integration of blockchain and AI technologies in a hybrid approach to secure IoT ecosystems. The AI methodology can continuously monitor the execution of smart contracts to detect suspicious

behaviors. Parameters are modelled to capture noticeable behaviors during the execution process. Whenever these parameters exceed predetermined boundary conditions, a smart contract will be flagged as potentially suspicious. Such parameters may include response time, execution cost, resource consumption, and the number of events occurring in a certain period. Cloud-based AI models have previously been used to detect anomalies in software system execution. This tackles the smart contract fraud concerns, securing the blockchains from such attacks. Blockchain technologies are able to guarantee the integrity of the audit logs required for AI model retraining [2].

### *B. Challenges and Limitations*

Despite the promising potential of the hybrid approach for enhancing IoT security, there are certain challenges that need to be addressed. While blockchain can enhance the security and integrity of data, one of its main challenges is scalability. As the number of IoT devices grows exponentially, the continuous addition of new blocks to the chain and the need for synchronization among all network nodes can lead to increased latency and inefficient processing of large amounts of data [2]. Another challenge is the high computational costs of the consensus protocols, especially for permissionless networks like Ethereum, which demands huge computational and storage resources. This can overwhelm the limited resources of IoT devices. Other challenges include the complex design of consensus protocols and the lack of regulation and guidelines regarding privacy metrics [4].

While AI can provide efficient solutions for various security-related issues, there are several challenges associated with its deployment in the IoT ecosystem. This includes the risk of model reverse engineering or stealing if AI-based algorithms are run on the edge or smart devices in the IoT ecosystem. Hackers/deep learning adversaries can exploit vulnerabilities in an AI model and create input samples to produce wrong outputs when tested on them (e.g., adversarial examples). This would circumvent the detection mechanisms, leading to false positives and security breaches. Implementing AI in IoT devices can also be challenging due to their limited computational resources. Intensive AI computations with a high number of parameters are expensive for low-power IoT sensors (e.g., wireless sensor networks). Additionally, it is difficult to safeguard the

confidentiality of locally trained AI models since the training datasets reside on the devices. Moreover, the black-box nature of AI models is also a concern for the IoT ecosystem.

## VI. THREAT DETECTION AND MITIGATION IN IoT ECOSYSTEMS

Within the frameworks of IoT security solutions, threat detection and mitigation play a critical role. From the perspective of cybersecurity, threat detection and mitigation primarily consist of two parts: the detection of potential security threats and the corresponding mitigation of these security threats according to pre-determined policies. Threat detection is defined as the process of identifying potential violations of security policies, where these violations can occur either due to external intrusion attempts or due to malicious behavior by insiders. In contrast, threat mitigation is defined as the process of taking necessary precautions or actions to eliminate the potential security threats as detected by the threat detection process. Threat detection and mitigation is common to any computer network and is thus indispensable to the security architectures of the IoT ecosystems. It is of critical importance to do threat detection and mitigation proactively, since any kind of security violation can have catastrophic consequences, especially with respect to unsafe control operations in cyber-physical systems [1]. Unlike existing computing technologies that are all largely homogeneous, there are a wide variety of different devices connected to the Internet within the IoT ecosystem. These devices encompass a diverse range of capabilities both in terms of hardware architectures and connectivity options. While this diversity forms the foundation of the IoT ecosystem, it also results in the proliferation of different challenges concerning any function to be universally deployed across the IoT ecosystem. Such challenges invoke the need for specially tailored solutions for securing IoT ecosystems to accommodate for the diverse range of connected devices and systems.

### *A. Importance of Threat Detection*

Within the Internet of Things (IoT) ecosystem, enhanced proactivity in the detection of threats is imperative as opposed to the current paradigms. IoT ecosystems comprise embedded devices (e.g., sensors and actuators) that communicate and interact semi-autonomously within the physical environment. The growth of IoT has led to a

proliferation of heterogeneous devices that (1) connect to the Internet and (2) expose data to other entities. However, this emergence of enormous volumes has furthered attack surface vulnerability and modified the incentive structures, hence resulting in new threats [1]. As these devices engage with the physical world, any IoT ecosystem attack can lead to hazardous consequences. In this vein, considerable potential dooms the emergence of IoT ecosystems as a new class of target for interactions with physical systems. The devastating potential of extant cyber-attacks on industrial control systems can be extrapolated to IoT environments.

In an IoT ecosystem, the inability to identify threats at an early stage can lead to damaging consequences, making IoT environment security a necessity. Public exposure renders IoT technology modifications prone to exploitation, resulting in anonymity, data integrity, confidentiality, and loss of embedded mobile systems. Understanding current threats using existing legacy technologies within an IoT environment is difficult. The traditional method of securing WAN and business sectors for securing the desktop system, LAN, or server does not suit mobile applications and embedded systems. Moreover, retraining of existing legacy technologies is proposed with huge investments and no efficiency guarantees.

#### *B. Common Threats in IoT Ecosystems*

This article aims to explore the pressing need for robust IoT security mitigation measures, with a focus on the novel approaches provided by blockchain and AI technologies. The notion of interconnected devices has existed for quite a while and was prevalent even before the inception of what is now known as the IoT or Internet of Things, and there are a range of attacks developed and implemented to pry over the security vulnerabilities of interconnected networks. Similarly, an overview of the common threats in an IoT ecosystem is provided to closely understand the set of compromised vulnerabilities for IoT devices and networks. Furthermore, a broad range of everyday security attacks and risks faced by IoT ecosystems from MPC to similar attacks is discussed.

IoT, or Internet of Things, is defined as a group of physical devices that are embedded with sensors, software and other technologies to exchange and gather data over internet. The idea behind IoT is to

connect a group of devices to the internet so they can work together, share information and complete certain tasks, without requiring human intervention or input. IoT can include an endless number of devices, ranging from everyday appliances like TVs and fridges to rubbish bins or industrial machines. Below is a set of common threats in an IoT ecosystem.

- Man-in-the-middle attack (MPC): An attack where an intruder secretly modifies the communication between two parties in an attempt to cheat both parties and redirect the flow of data.

- Replay attack (RE): An attack that occurs when a malicious user captures a valid data transmission and is able to impersonate the victim user by replaying the captured data at a later time.

- Link spoofing attack (LS): A denial-of-service attack where a malicious IoT node creates false links in the network to IoT nodes by connecting with them, thereby creating a false routing table that influences the configuration of IoT nodes.

- Physical attacks (PA): Attacks that involve the physical tampering of a device with intent to extract confidential keys or inject malicious codes to the device.

- Eavesdropping attack (EA): Attacks that occur when intruders listen to communications either by using a backdoor or through illegitimate access to the network.

- Denial-of-service attack (DoS): Attacks that prevent the IoT system from functioning properly for its intended users through service overload by flooding the network with numerous transmissions [1].

- Replay attack (RE): Similar to passive TS, but in this case, the replayed TS is altered in a controlled manner in an attempt to trick the system into a fault state (not handled by valid models). Further, the total number of transmissions is reduced with respect to a valid usage, to accelerate the time lapse for errors to appear (i.e. lower the probability of verification).

- Distributed denial-of-service attack (DDoS): Similar to DoS attacks, but in this case the attack is launched from a set of compromised nodes in large scale. This makes the detection of the malicious behavior even more difficult [2].

- Sybil attack: An adversary claims multiple identities and injects false data to the system from different nodes at the same time, magnifying the impact of the attack. In this case, the malicious

nodes operating under different identities are known to the adversary, along with their personal data.

- Insider attack: One or several nodes in the network, fulfilling the role of normal users, are compromised by the adversary. This node or set of compromised nodes work under the attacker's control and, hence, have knowledge of the system preventing the detection of attacks.
- Spoofing: An illegitimate node attempts to impersonate a legitimate one in order to connect and inject malicious traffic to the system, compromising both the integrity and the authenticity of the transmitted data.

## VII. CASE STUDIES AND IMPLEMENTATIONS

In the realm of the Internet of Things (IoT), data is paramount. The companies have come to realize that they now own more data than they ever dreamed of. Sophisticated robots running machine learning algorithms produce performance and condition data. Vehicles with multiple sensors share position, speed, weather, and potential accidents with other cars and the municipalities running traffic lights and road systems. Buildings containing thousands of control points generate data on comfort, heat, humidity, energy consumption, and parking availability. Parks, forests, and other natural habitats are continuously monitoring temperature, pollution, and moisture levels [1]. Above all this data are a plethora of personal devices gathering information about their users' whereabouts, movements, expenses, and insulin levels.

With the right algorithms and sufficient computing power, all this data can be used to learn and understand the physical world. At the same time, the data can also be weaponized. Data breaches create paranoia; behavior can be predicted, and users manipulated; preemptive strikes can be automated and game rules changed. Public intellectuals have warned against the surveillance state run by big corporations. Security concerns have also been raised regarding rogue states using similar mechanisms for subjugation [3]. Lawmakers are trying to define data ownership, hoping to reverse the monopoly by big companies. The concern is that politically and socially charged data cannot be protected unless it is impossible to collect. This means reimagining how the data is generated. Now, a scenario where simple

electronic device identities are reused is proposed instead of limited IP address scenarios.

### A. Real-World Examples

This section showcases real-world examples of successful implementation of IoT security using blockchain and AI technologies. The case studies focus on tangible implementations of the explored technologies, with learnings from these real-world deployments. Its intentions are to provide a contextual understanding of the technologies, accompanied by examples of how they have been used to solve problems in IoT securely.

This case study, published in 2021, is based on a real-world blockchain and AI implementation on a smart biogas plant. The IoT-based biogas installation remotely monitors temperature, humidity, voltage, gas flow, and load conditions using sensors and controllers. An artificial neural network (ANN) model predicts gas production based on past data. However, the biogas production process is sensitive, requiring high security. Using blockchain technology is proposed as a solution, ensuring a secure environment for IoT devices, protection against data manipulation, and decentralized control. Learning is provided on how to harness blockchain technology to provide a CORAL proof-of-concept for IoT-based industrial use cases, alongside the lessons from the implementation [1].

### B. Best Practices

This section outlines the lessons learned through case studies of successful IoT security applications using blockchain and AI, providing a collection of best practices and key learnings. These measures have been put to the test in various industries as early adopters of security measures for IoT devices. Furthermore, they present a demonstration of innovative security measures for IoT, although they are not widely implemented yet, state the hazards, and suggest improvements. The first group highlights businesses that have adopted and enhanced accountability, privacy, confidentiality, integrity, or availability measures. The second category discusses efforts in progress to improve the level of protection. These are large organizations that have begun experimenting with blockchain-based IoT solutions. Since the demonstrations are not fully operational and implemented by only a few companies, hackers are not executing the proposed attacks to bypass these

innovative security measures [3]. Initially, the best-known security problems were conducted for assets rather than information. Fast transportable valuables such as gold, cash, and diamonds were the first targets when violent crime was not performed within a nation. This developed into dirty money laundering operations without having a direct connection to a bank by avoiding government controls and taxes. Access to black-markets became possible on the internet, where pharmaceuticals, drugs, and weapons could be transferred outside the law for large sums of money [1].

#### VIII. SCALABILITY AND PERFORMANCE CONSIDERATIONS

Measured by the number of connected devices, the internet of things (IoT) is the most extensive network comprising wireless sensor nodes connected via internet providing real-time information to one another. This project proposes a new Lightweight Intelligent Trust-based End-to-End Security Approach for IoT Network using Artificial Intelligence (AI) and blockchain technology and thus upholding the significance of AI and blockchain technology in designing vulnerability-free security for IoT networks.

The industrial internet of things (IIoT) will connect all or individual devices using machine-to-machine (M2M) communication technology in a wide area. All the sensors are connected to smartphones, computers, servers, cloud, gateways, base stations, etc., providing services like surveillance, healthcare services, automotive, automotive's controlling, and many other applications. With rising use of IIoT or connected devices expectable cyber-attack incidences have also increased causing monetary loss, injury, even deaths. Thus vulnerability-free and easy-to-implement end-to-end security for the IoT network has become critical.

##### A. Scalability Challenges in IoT Security

In recent years, blockchain and AI technologies have received great attention due to their potential to provide innovative solutions to several security problems in IoT ecosystems. This solution proposed to detect, log, and mitigate various attacks in industrial IoT by employing blockchain and machine learning techniques. Their model comprises three stages of processing. At the first stage, the collected data about a potential attack is prepared in the proper format for further

processing. Benign and attack data are used for deep learning training, and the model is executed. The output model is then used to infer newly arrived data. Depending on the model's output, the second stage is conducted either to record the event or to take immediate legitimacy measures to block the attacker. Detected attacks are recorded on the local blockchain in the final mitigation stage, with the possibility of additional logging of recorded blocks on the global blockchain [1].

This solution highlights the challenges and hurdles in the implementation of the proposed model, with special consideration for scalability. Although there are tests with different model configurations, the full output of this algorithm is still too slow for real-time applications. Therefore, further examination of the proposed machine-learning model is required in order to find the optimal architecture that would result in both the highest possible accuracy and performance. As mentioned previously, the biggest bottleneck of the application on the fully decentralized architecture is the overhead that occurs during the transaction verification and the mining processes due to increasingly large datasets and their rapid growth with time, which is expected to grow exponentially in the future [3].

##### B. Performance Metrics

Performance metrics and considerations are very relevant in measuring the efficiency of the proposed mechanism designs and solutions for IoT security. To efficiently assess different promising IoT security solution designs and implementations, it is necessary to first define the context of their measuring relevant performance metrics and measurements, including how and what they are best exploited or tackled at specific statistical or computer benchmarks.

- **Scalability & Latency** - After ensuring that privacy and integrity requirements are satisfied, performance is the primary concern for the proposed countermeasures. It is meaningful to represent how the execution and validation latencies vary concerning key metrics such as the network size. Attacks against the nodes and communication links, including denial-of-service attacks and jamming, which are highly relevant in large IoT networks, should also be measured. Scalability regarding the number of skimming rounds or followers, as well as the number of security controllers or schemers, should also be

part of performance validation efforts and is expected to be highly suitable in large IoT networks.

- **Storage security** - This performance evaluation approach is easily conceptualized when there is an external threat of having cameras or wireless nodes physically overtaken. A large body of research exists on how to secure the physical implementation of an IoT node. Key settings such as barriers or tamper and water-resistant coatings can be considered by default in the performance evaluation of the proposed solutions. Such key settings can then define an IoT node or camera as robustly secure or relatively weakly secure. It also applies when having an external threat of a cloud forensics audit is possible. In this case, cloud FT can be considered as an affront.

#### IX. RESOURCE CONSTRAINTS AND OPTIMIZATION

The IoT security using blockchain and AI approaches suffers the inherent constraints of any IoT security approaches: limited bandwidth, limited processing power, limited battery, limited storage, and very limited interoperability. The on-device AI and edge computing reduce data transmission overload and latency but require thorough management of hardware resources such as CPU, GPU, TPU, memory, network, etc. As a result, it depends on the service type, the quality commitment, and the service distribution. Resource management policies can be operation models run by a business or load-balancing algorithms to distribute on-device and edge computing resource usages. However, how to use them in a multi-tiered IoT AI blockchain security and trust model is a gap in the literature. Most importantly, how to mitigate the Coase theorem: the cost-free market arbitrage, where the edge computing constantly drains on-device resources or vice versa, is an upper-studied issue [1].

Inspired by the Smart \$-Contract protocol, a novel resource planning and timing policy is proposed based on multiple reinforcement learning (RL) techniques: Long-Short-Term Memory (LSTM) Neural Network and Q-learning. These models facilitate internal mechanisms to adaptively learn load-balanced resource consumption in the AI blockchain security framework. A multi-agent-safe training policy with differential evolution agents to avoid market rivalry is designed. Moreover, upper bounds to aggregate hardware resources in a competitive environment allow fair distribution of

earnings between AI models and, thus, a fair resource usage to enterprises. It is worth noting that the problem of directly mapping non-convex safety requirements on RL agents is still an open question [4].

##### A. Resource Management Strategies

Resources are essential for any successful implementation of blockchain-enabled security systems in IoT scenarios. Resources include the storage capacity for data blocks, computational power for smart contracts, IoT devices, and network bandwidth. All these resources significantly impact the performance of blockchain-based IoT solutions (with regards to various application-dependent parameters like latency, energy usage, accuracy, etc.). There are two crucial aspects of resource management in this context:

A case study of a learning-based method for resource allocation is presented. With regard to a certain IoT application, the method incurs negligible overhead in terms of processing time and data. Most importantly, the performance of the application (in general, accuracy of ML models) increases with the resource allocation [1]. On the other hand, the accuracy of the application does not decrease drastically with the lowering of resource availability for blockchain operations used in the application (smart contracts for executing the services). A comprehensive plan for IoT integration with Energy Efficient Blockchain and Localization Algorithm (blockchain-AI algorithm and resource allocation machine learning technique) processes is presented, covering different areas from implementation to optimization.

Smart contracts and IoT need several resources for their operations to excel, smart contracts need gas and devices, network bandwidth and storage at the nodes. Proper resource management strategies are also required to overcome the constraints and challenges of factors degrading the implemented solutions in a resource-constrained domain. Here, the models of IoT devices' accessibility and load on the network are explained. Then the modeling of these constraints with respect to resource allocation for AI and blockchain supported by resource provisioned smart contracts transactions and model execution respectively [5].

### B. Optimization Techniques

Optimization techniques specific to the objective of improving the performance of IoT security measures based on blockchain and AI technologies can be broadly categorized into general optimization techniques and those specific to IoT security measures. A systematic review of published works pertaining to the aforementioned categories of optimization techniques and their implementations is provided below. General optimization techniques are covered first, followed by optimization techniques specific to IoT security measures.

**General Optimization Techniques** A novel framework that integrates IoT, smart contract, and AI features to anticipate safety risks in smart manufacturing and optimize preventative actions to minimize risks is presented. Additionally, a detailed description of optimization techniques that can be used to solve risk minimization problems is provided. Industrial internet of things (IIoT) is an innovative system that integrates manufacturing equipment, production control systems, and enterprise information systems. It aims to provide real-time and transparent data for decision-making across the entire value chain of manufacturing. However, it is vulnerable to cyberattacks due to its reliance on the internet. Deep reinforcement learning (DRL)-based intelligent decision-making approaches can recognize hazards, evaluate safety risks, and recommend proposed actions to decision-makers. Safety Control Logic is encoded as a semi-Markov decision process for risk minimization with respect to the level of enforcement (L1~L3), which is only a fuzzy set of probabilities associated with the state of the system being safe. Risk mitigation plans must take into account both temporary remedial measures (such as L1) and long-term solutions (such as L2 or L3). The hazard and fault tree models are coupled with Markov processes to translate the system model into a discrete-time and continuous-time framework [1].

A novel deep reinforcement learning-based algorithm that transforms the recovery decision process from an MDP to a variable-probability Markov decision process (VP-MDP) is proposed. It simultaneously learns winning strategies for an attacker and a defender with regard to the same state (the post-attack state). The defender's strategy is hidden in the VP-MDP, and a probabilistic

strategy for recovering from cyber-physical attacks is inferred. To the best of the authors' knowledge, a game-theoretic framework in which a defender's sequential recovery actions in CPS are modeled by a variable-probability MDP is presented for the first time. The variable probabilities are successively optimized to maximize the defender's expected utilities. Since the successfully recovering state space is usually unknown to the defender in a post-attack scenario, a deep reinforcement learning-based algorithm is required to derive the recovery strategy [5].

Modeling cyber-physical attacks enables the simultaneous evaluation of the effects of attacks on system dynamics and the attack consequences for defenders, including attackers, and defenders' recovery actions. To allow a comprehensive investigation of the attacker-defender interactions in CPS, a multi-entity attack-defense framework is established. The limited knowledge of both an attacker and a defender is taken into account to optimally design the defender's longest recovery strategy in the real world. An-attack recovery problem is defined in a game-theoretic framework with a hybrid model that combines a variable-probability Markov game with a stochastic hybrid system. This problem is formulated as a mathematical optimization problem, which is non-convex and intractable for attackers and defenders because of massive action spaces.

### X. FUTURE DIRECTIONS AND EMERGING TRENDS

The convergence of AI and blockchain in IoT security represents a dynamic and rapidly evolving domain. Future directions include innovations in blockchain consensus algorithms, resource-efficient AI applications, federation mechanisms, and analysing sector-wise requirements for integrated models. Several challenges accompany such promising advances, including the adaptation of blockchain to AI techniques, development of lightweight blockchain players, enhancement of prediction quality, and analysis of the cost to efficiency ratio for federated AI models. This lays the groundwork for the proposal of dedicated areas of R&D activity to act as a basis for future dialogues among all actors of the envisaged smart and connected world [1].

There is potential for the emergence of common standards in the interaction between AI and blockchain in IoT applications, thereby facilitating genuine scalability of blockchain-based AI

systems. In this engagement, the focus needs to be placed on the growing number of AI applications of various complexities interacting with IoT devices and networks. There remain significant challenges regarding both the incentivization of actors sharing knowledge and the availability of trustworthy and quality-assured data of all kinds [3].

#### A. *Advancements in Blockchain and AI Integration*

The integration of artificial intelligence (AI) with blockchain technology has gained momentum during the past few years, as witnessed by the growing number of publications and academic research projects [3]. The arrival of 5G, edge computing, fog computing, and low-cost sensors has heralded the birth of new smart applications that rely on the Internet of Things (IoT) for efficient and timely services. Yet, with the mass-generation and demand for smart information, smart applications are increasingly vulnerable to privacy and security threats, particularly when users' sensitive smart data is shared with third parties, or when these smart applications are under attack. Consequently, blockchains, with their distinctive decentralized design and tamper-resistant capabilities, have been adopted for a growing number of smart applications for securing smart data transactions and communications. Nevertheless, in smart businesses or societies densely populated with objects or devices, it remains progressively more challenging to adequately store massive data generated from many sources of devices without compromising data integrity [2]. In parallel, AI has become a necessity for smart applications in the age of IoT due to its capability for information processing after sensing or gathering information in a smart environment. It has been extensively adopted to increase business productivity or users' utility in smart applications by discovering valuable or interpretable knowledge or patterns. Cloud computing, as a pivotal enabling technology for IoT, has fostered the widespread commercial adoption of AI in smart applications due to its potential for keeping the enormous power-consuming or complicated processing resources in a centralized cloud environment. Similarly, the arrival of the Sybil era has heralded a new form of architecture in which various low-cost, low-power, and wide-domain sensors are embedded in many everyday objects. However, such cloud-based AI

processing in IoT environments incurs privacy and security issues due to the potential over-sharing of datasets or information to clouds about users' sensitive or personal experiences.

#### B. *Potential Innovations in IoT Security*

The convergence of blockchain and AI technologies in the realm of IoT security presents a myriad of innovative possibilities. Firstly, smart contracts can be empowered with AI algorithms to establish a proactive and adaptive security means. By continuously learning from new attacks, such smart contracts can automatically modify their operations, thus guaranteeing defense adaptability as new threats emerge. These contracts can also minimize false alarm incidents through AI/ML-driven postures. Additionally, the increasing availability of large-scale IoT data provides fresh possibilities for the use of AI technologies in blockchain structures to intelligently handle transaction traffic and speeds [1]. The mixture of AI techniques within the network of peer nodes can enhance scalability and delay in transaction validation through intelligent predictions concerning network congestion levels.

Moreover, the integration of federated learning systems with blockchain technologies can enable non-intrusive machine learning activities. As federated learning models concentrate on the training of AI models in a distributed manner, only hyperparameters are shared through the blockchain. Such methods can prevent sensitive information from being stored on a single database, thus ensuring the compliance of personal data (GDPR) regulations [2]. In addition, distributed ledger technologies may provide IoT devices and networks with native digital currencies as an incentive for good and reliable behavior. Sometimes, micro-payments can also be integrated into smart contracts as an alternative to the traditional fine imposed for malicious behavior. Such payments may include disclosing monetary penalties, rewards, or fines, thus ensuring scalable mitigation mechanisms.[5]

### XI. CONCLUSION

#### A. *Summary of Key Findings*

A comprehensive summary of the key findings and insights derived from the exploration of IoT security using blockchain and AI is presented, encapsulating the core takeaways and contributions of the essay [1]. The growing importance of the

Internet of Things (IoT) as a key technological trend is discussed, along with various IoT applications across domains. It is outlined that AI-driven solutions can provide a significant edge in addressing IoT issues and building new applications [3]. The challenges of IoT networks brought about by heterogeneous devices, different protocols, and isolated architectures are characterized. This is followed by a discussion of security issues in the context of analysis on trust, privacy, data integrity, and malicious attacks.

The need for a robust dynamic approach using blockchain and AI for security enhancement at multiple IoT layers is established. Moreover, a blockchain framework is proposed to be deployed in conjunction with AI in IoT applications. Different integration approaches of blockchain and AI technologies are outlined after an analysis on their effectiveness across IoT layers. This assignment can be a first step toward boosting the resilience of IoT application security using a blockchain-AI dynamic strategy. A reflective synthesis of the key findings to provide a comprehensive understanding of the implications and potential applications of the research is offered. Various applications of IoT, the advantages and risks of integrating blockchain and AI into IoT security, the challenges in the integration process, and points for further research and improvement in IoT security are also discussed.

#### *B. Implications and Recommendations for Future Research*

One of the most important steps one can take in their research career is to outline the directions for future research and the potential implications of one's findings. Not only does this demonstrate an understanding of the feeling position of one's work in a broader field, but fleshing out the implications of one's research presents one's findings as a springboard for further investigation and can lend weight to one's claims with respect to the importance of the findings presented. The research efforts presented here within this thesis are consequently extended with the following implications and recommendations for future research. Avenues for investigation along the following lines are delineated on a chapter-by-chapter basis. Academic and social implications and issues arising from the investigations that take on a theoretical or hand-in-hand with environmental considerations are also discussed.

The findings of this thesis imply that blockchain technologies can contribute to the security and performance of IoT systems, in particular in terms of intrusion detection, trust establishment, anomaly detection and prevention. However, it is evident from a consideration of the work presented in Chapters 4, 6 and 10 that the performance improvement offered by the blockchain AI techniques could exceed that of traditional non-blockchain AI approaches in certain IoT use cases. Further investigation of the potential additional performance benefits that the inclusion or exclusion of blockchain technologies might have with respect to a selection of machine learning algorithms in the security approach in a wider variety of IoT use cases is therefore warranted [2].

XIII. REFERENCES:

- [1] P. Bothra, R. Karmakar, S. Bhattacharya, and S. De, "How Can Applications of Blockchain and Artificial Intelligence Improve Performance of Internet of Things? - A Survey," 2021. [\[PDF\]](#)
- [2] S. Selvarajan, G. Srivastava, A. O. Khadidos, A. O. Khadidos et al., "An artificial intelligence lightweight blockchain security model for security and privacy in IIoT systems," 2023. [ncbi.nlm.nih.gov](https://ncbi.nlm.nih.gov)
- [3] A. Rahman, D. Kundu, T. Debnath, M. Rahman et al., "Blockchain-based AI Methods for Managing Industrial IoT: Recent Developments, Integration Challenges and Opportunities," 2024. [\[PDF\]](#)
- [4] K. Prasad Satamraju and M. B., "Proof of Concept of Scalable Integration of Internet of Things and Blockchain in Healthcare," 2020. [ncbi.nlm.nih.gov](https://ncbi.nlm.nih.gov)
- [5] A. Sasikumar, L. Ravi, K. Kotecha, J. R. Saini et al., "Sustainable Smart Industry: A Secure and Energy Efficient Consensus Mechanism for Artificial Intelligence Enabled Industrial Internet of Things," 2022. [ncbi.nlm.nih.gov](https://ncbi.nlm.nih.gov)

# Large-Scale E-Commerce Product Selection Using Skyline Queries in Heterogeneous Computing Environments

Walid Khames

*Aeronautical and Spatial Studies Institute University Blida1 Algeria*

khames.walid@univ-blida.dz

**Abstract**—In this paper, we introduce an efficient skyline algorithm designed for large-scale data management in heterogeneous computing environments. This algorithm use both index-based and sort-based approaches to optimize skyline computation over E-Commerce data streams. Our parallel implementation harnesses the computational power of Graphics Processing Units (GPUs) to accelerate skyline queries, significantly outperforming traditional CPU-based methods. Optimized for real-time e-commerce product selection and other data-intensive applications, our approach ensures continuous skyline updates under the count-based sliding window model. By exploiting the massive parallelism of GPUs and efficient indexing and sorting techniques, our methods achieve substantial improvements in performance, scalability, and responsiveness. Extensive experiments conducted on real-world and synthetic datasets demonstrate that our proposed algorithms provide superior efficiency compared to baseline skyline techniques such as BNL, BskyTreeP, BskyTreeS, Salsa and SFS. Our findings indicate that the combination of GPU acceleration with advanced sorting and indexing strategies presents a powerful solution for large-scale skyline queries in big data environments.

**Index Terms**—Data Management, Big Data, Multicore Architecture, Continuous skyline, Parallel computation, Data Stream.

CPU to accelerate skyline computation. However, its performance is constrained by limited memory bandwidth and the inherent parallelism restrictions of CPU architectures. To overcome these limitations, we propose ECSQ algorithm, a GPU-accelerated algorithm that exploits massive parallelism to enhance skyline computation for Large-Scale E-Commerce Product Selection. While some studies have explored GPU-based skyline computation [11]–[15], most focus on static datasets rather than continuous streaming scenarios. Our work bridges this gap by designing an efficient GPU-based continuous skyline algorithm.

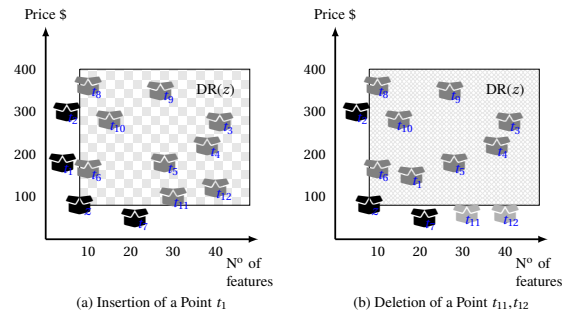


Fig. 1. Continuous Skyline query with new and expired tuples.

## I. INTRODUCTION

Skyline queries are fundamental in multi-criteria decision-making [1], widely applied in recommendation systems [2], decision support [3], data mining [4], query optimization [5], and traffic monitoring [6]. Unlike traditional queries that return exact matches, skyline queries identify non-dominated data points, helping users find optimal trade-offs across multiple conflicting attributes.

A continuous skyline query (CSQ) extends this concept to dynamic data streams, where tuples continuously arrive and expire based on a sliding window mechanism. Figure 1 illustrates how a CSQ maintains the skyline set by incorporating new tuples while discarding expired ones. Additionally, Figure 2 demonstrates the count-based sliding window model, where only the most recent  $Z$  tuples are considered at any time.

Handling skyline queries in E-Commerce data streams is computationally challenging due to the curse of dimensionality, leading to an exponential increase in skyline points. Several CPU-based algorithms have been proposed [7]–[9], but they often struggle with E-Commerce streaming data. Our previous work, Parallel RSS (PRSS) [10], utilized multicore

Skyline queries assist in multi-criteria decision-making by selecting a subset of optimal objects that are not dominated by any other object in the dataset. This is useful in e-commerce, financial analysis, and traffic monitoring. Table I illustrates how skyline queries help in selecting the best products based on attributes such as price, user rating, shipping time, return rate, stock availability, and profit margin. A product is part of the skyline if no other product is strictly better in all dimensions.

The skyline products (marked as "Yes" in the last column) represent the best trade-offs based on user preferences. However, as datasets grow in size and dimensionality, skyline computation becomes increasingly complex. To address these challenges, our ECSQ algorithm uses GPU acceleration to perform skyline computations efficiently in real-time E-Commerce data streams.

The rest of this paper is organized as follows. The next section reviews related work on skyline computation, discussing both CPU-based and GPU-accelerated approaches. Then, we

TABLE I. E-commerce Product Selection with Skyline

Product	Price	User Rating	Shipping Time	Return Rate	Stock Availability	Profit Margin	Skyline
P1	30	0.9	2	5	500	40	Yes
P2	25	0.85	3	8	600	35	No
P3	20	0.92	1	4	300	50	Yes
P4	28	0.88	2	6	400	38	No
P5	22	0.9	3	5	550	45	No
P6	18	0.91	1	4	250	48	Yes
P7	35	0.87	2	7	450	42	No
P8	24	0.89	3	6	500	40	No
P9	19	0.93	2	5	200	52	Yes
P10	27	0.86	2	6	400	39	No

formally define the problem and outline the challenges of E-Commerce skyline queries in continuous data streams. After that, we present our proposed GPU-based Skyline (ECSQ) algorithm, detailing its design and implementation. We then evaluate the performance of ECSQ through extensive experiments and comparisons with existing methods. Finally, we conclude the paper with a summary of our findings and potential future research directions.

## II. RELATED WORKS

Within this section, we first introduce the symbols and terminology used throughout this paper. We then review relevant literature and related works to provide a comprehensive context for our study.

### A. Skyline Queries

Since the introduction of the skyline operator by Börzsönyi et al. [16], skyline queries have become a fundamental topic in database research, offering an effective approach to extracting interesting objects from multi-dimensional datasets. In this context, database tuples are treated as multi-dimensional data points, and the skyline query identifies the most relevant ones based on user preferences, without relying on cumulative functions.

**Definition 1 (Dominance [8]):** Given a tuple  $x$  in an  $\mathcal{N}$ -dimensional space, we denote its value at dimension  $i$  as  $x[i]$ , where  $1 \leq i \leq \mathcal{N}$ .

For two tuples  $x$  and  $x'$ :

- $x[i] < x'[i]$  denotes that  $x[i]$  is better than  $x'[i]$ .
- $x[i] \leq x'[i]$  means that  $x[i]$  is either better than or equal to  $x'[i]$ , i.e.,  $(x[i] < x'[i]) \vee (x[i] = x'[i])$ .
- If  $x[i] < x'[i]$ , then it follows that  $x[i] \leq x'[i]$ .

A tuple  $x$  dominates another tuple  $x'$ , denoted as  $x < x'$ , if and only if:

- 1) For every dimension  $1 \leq i \leq \mathcal{N}$ ,  $x[i] \leq x'[i]$ .
- 2) There exists at least one dimension  $1 \leq m \leq \mathcal{N}$  such that  $x[m] < x'[m]$ .

We use  $x \not< x'$  to indicate that  $x$  does not dominate  $x'$ , and  $x \approx x'$  to express that  $x$  and  $x'$  are incomparable, meaning neither dominates the other:

$$(x \not< x') \wedge (x' \not< x).$$

These relations extend naturally to sets of tuples:

- $x < X$  means that  $x$  dominates every tuple in  $X$ .

- $x \approx X$  means that  $x$  is incomparable with every tuple in  $X$ .

- **Incomparability [8]** Two  $\mathcal{N}$ -dimensional tuples  $x, x' \in \mathcal{T}$  are considered incomparable within  $\mathcal{T}$  if neither dominates the other, denoted as  $x \approx x'$ . Formally:

$$x \approx x' \iff x \not< x' \wedge x' \not< x.$$

This property is crucial in determining whether a tuple belongs to the skyline set since a skyline tuple must be incomparable to all other tuples in the dataset.

- **Continuous Skyline [8]** Given a continuous  $\mathcal{N}$ -dimensional dataset  $\mathcal{T}$ , a tuple  $x \in \mathcal{T}$  is a skyline tuple if no other tuple  $x' \in \mathcal{T}$  dominates it, i.e.,

$$\mathcal{M} = \{x \in \mathcal{T} \mid \nexists x' \in \mathcal{T}, x' < x\}.$$

The skyline set  $\mathcal{M}$  consists of all tuples that are not dominated by any other tuple in  $\mathcal{T}$ .

Skyline queries have been widely studied and applied across various domains, demonstrating their effectiveness in multi-criteria decision-making. These applications include environmental monitoring [17], IoT [18], aviation industry optimization [19], [20], and social media analysis [21].

In general, skyline algorithms can be broadly classified into two categories [22]:

- **In-core algorithms:** These algorithms are designed to compute skyline queries on datasets that reside entirely in main memory, ensuring fast processing times. Notable examples include methods such as *G-Skyline* [23], *SFS* [24], and the foundational *BNL* algorithm [16].
- **Out-of-core algorithms:** These algorithms are tailored for handling large-scale datasets that do not fit in main memory, requiring access to secondary storage. To improve efficiency, they employ specialized techniques for managing disk-based data retrieval and processing. Out-of-core approaches can be further categorized as follows:
  - **Index-Free Techniques:** Methods that do not rely on indexing structures for dataset  $\mathcal{T}$  are referred to as "index-free" techniques. Examples include *BNL* and *D&C* [16], *Iskyline* [25], *VP* and *KISB* [26], as well as Object Space Partitioning (*OSP*) [27]. While these approaches eliminate the overhead of maintaining an index, they tend to incur higher computational costs due to operations such as presorting and direct pairwise comparisons between tuples. Consequently, index-free methods often suffer from performance bottlenecks in large-scale datasets [22].
  - **Index-Based Techniques:** Index-based methods use data structures to efficiently retrieve relevant tuples while minimizing unnecessary comparisons. By constructing an index on specific attributes, these algorithms can significantly speed up skyline computation [28]. Several approaches use spatial indexing structures such as *R-trees* [29], [30] and *B-trees* [31]–[34]. Other techniques include trie-based indexing like *SkyMap* [35].

Despite their efficiency, index-based approaches face challenges such as being limited to certain data types

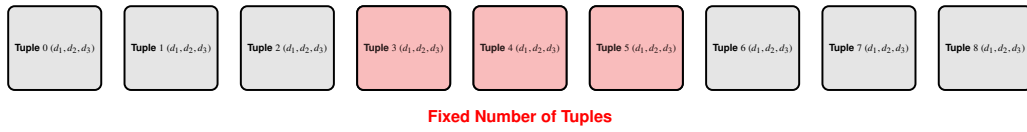


Fig. 2. Count-Based based Sliding Window.

or incurring high indexing costs, which may offset their benefits. Additionally, dominance-based skyline techniques have been developed to reduce pairwise comparisons, maintain skyline structures using trees or lattices, and exploit tuple incomparability to eliminate redundant evaluations. Examples of such methods include *BSkyTree* [36] and *BJR-tree* [37].

### B. Skyline Computation in E-Commerce and Recommendation Systems

Skyline computation has emerged as a pivotal technique in multi-criteria decision-making, particularly in domains like e-commerce and recommendation systems, where users seek optimal trade-offs among competing attributes. Over the years, researchers have proposed various algorithms and optimizations to enhance skyline query efficiency, scalability, and applicability to different data types.

The groundwork for skyline queries in web information systems was laid by Balke and Güntzer [38], who introduced techniques for processing categorical data, a common yet challenging data type in e-commerce. Their work addressed the limitations of traditional skyline algorithms, which primarily focused on numerical attributes, by proposing methods to handle discrete, non-ordered domains. The experiments primarily used synthetic datasets with varying distributions (correlated, anticorrelated) to assess robustness.

Building on this, Feng et al. (2009) [39] proposed the Rainbow Product Ranking algorithm, which integrates skyline computation with product ranking in e-commerce. Their approach used real and synthetic datasets to simulate user preferences, emphasizing the need for dynamic query processing to adapt to changing user interests. The study did not specify the programming language used but highlighted the importance of indexing techniques to accelerate query responses.

Recognizing that users often struggle to articulate precise preferences, Lofi et al. (2010, 2011) [40], [41] introduced example-based heuristics to refine skyline queries interactively. Their work focused on dynamic, user-guided preference elicitation, where partial inputs from users iteratively narrow down the skyline. The experiments employed real e-commerce datasets to validate the heuristic's effectiveness, demonstrating improvements in recommendation relevance. The processing was centralized, relying on sorting and filtering to maintain real-time responsiveness.

As datasets grew in size and complexity, researchers shifted toward distributed and parallel processing. Chung et al. (2012) [42] tackled combinatorial skyline queries, where the goal was to find optimal combinations of items rather than individual products. Their algorithm employed partitioning and pruning strategies to reduce computational overhead, tested

on synthetic datasets with varying dimensionality. The study emphasized scalability, measuring query response time and memory consumption as key metrics.

Similarly, Sessoms and Anyanwu (2013) [43] introduced SkyPackage, a framework for finding optimal packages of items on the Semantic Web. Their approach extended skyline computation to semantic relationships, requiring specialized indexing (e.g., RDF-based structures) for efficient retrieval. The experiments used real-world linked data, underscoring the challenges of incomplete and uncertain datasets in web environments.

Personalization became a central theme in later works. Das Sarma et al. (2014) [44] explored diversified and personalized product search, integrating skyline queries with user behavior modeling. Their experiments on real e-commerce logs demonstrated the trade-off between personalization and diversity, evaluated via precision and recall metrics.

Uncertainty-aware methods also gained traction. Zhou et al. (2016, 2018) [45], [46] introduced probabilistic skyline queries to handle uncertain product ratings, proposing a Top-k Favorite Probabilistic Products (TFPP) algorithm. Their work used synthetic and real datasets with injected uncertainty, measuring accuracy and computational efficiency. Parallely, Islam et al. (2017) [47] and Yin et al. (2018) [48] investigated reverse skyline queries to identify influential products or prospective customers, employing grid-based indexing to speed up dominance checks.

Recent advancements have focused on real-time and parallel processing. Tai et al. (2021) [49] proposed a dynamic skyline algorithm with parallel execution to identify profitable products under promotions. Their experiments, conducted on cloud-based systems, demonstrated significant speedups using MapReduce and GPU acceleration, with throughput and latency as critical benchmarks.

Recently, Khames et al. introduced PRSS (Parallel Range Search Skyline), a E-Commerce skyline algorithm tailored for sliding window-based skyline computation on multicore processors [10]. PRSS enhances dominance tests and maintains skyline results dynamically, achieving substantial speedup on real-world and synthetic datasets. Despite these advancements, real-time processing of large-scale streaming data remains a challenge, necessitating further research into GPU-based solutions.

The evolution of skyline computation reflects a shift from static, centralized methods to dynamic, distributed, and uncertainty-aware techniques. Key algorithmic strategies—such as indexing (R-trees, grid-based), partitioning, and parallel processing—have been instrumental in improving scalability. Performance evaluations consistently highlight query response time, memory efficiency, and accuracy as decisive

factors.

Future research may explore deep learning-enhanced skyline queries for preference modeling and federated skyline computation for decentralized data. Additionally, explainable skyline recommendations could bridge the gap between algorithmic efficiency and user trust in e-commerce systems.

Existing GPU-based skyline computation techniques demonstrate significant performance gains; however, a notable gap remains in effectively handling real-time data streams with high dimensionality. Our proposed GPU-based skyline algorithm, *ECSQ*, aims to bridge this gap by introducing a High Dimensional Search Skyline algorithm designed for E-commerce product selection. The next section will detail the characteristics and advantages of the *ECSQ* algorithm.

### III. PROBLEM DEFINITION

This section introduces the notations and terminology used throughout this paper, defines the challenges of continuous skyline computation, and presents our GPU-based solution, *ECSQ*.

**Challenges in Continuous Skyline Computation:** Skyline computation identifies non-dominated points based on multiple criteria, aiding decision-making applications. In continuous data streams, where new points arrive dynamically, maintaining skyline results in real time is challenging. Traditional algorithms designed for static datasets struggle with large-scale, high-velocity streaming data.

To address these issues, *ECSQ* employs a *count-based sliding window model*, using GPU parallelism to process large data volumes efficiently. The core objectives of *ECSQ* are:

- 1) **Real-time Responsiveness:** Achieve near-instantaneous skyline updates in dynamic streams.
- 2) **Scalability:** Utilize GPU parallelism for handling large, ecommerce datasets.

By achieving these goals, *ECSQ* provides a robust solution for continuous skyline computation.

#### A. GPU-Based Skyline Computation

*ECSQ* incorporates *dimension indexing*, where dataset dimensions are sorted to optimize skyline queries. This indexing minimizes dominance comparisons, significantly improving performance.

With GPU acceleration, dimension indexing and skyline computations are parallelized across thousands of threads. Each GPU thread processes a subset of tuples, enabling parallel exploration and dominance checks, as illustrated in Figure 3.

**Definition 2:** (GPU Dimension Index) Given an  $\mathcal{N}$ -dimensional dataset  $\mathcal{T}$ , the *dimension index*  $A$  consists of sorted collections of index entries. Each entry contains:

- A *header pointer* referencing tuple metadata.
- A *dimension value* storing the attribute's value.
- A *dimension pointer* linking to the next dimension.

Entries within a dimension  $A_i$  ( $1 \leq i \leq \mathcal{N}$ ) are sorted based on a predefined order.

**GPU Parallelization in ECSQ:** *ECSQ* extends our previous PRSS [10] algorithm by using GPU parallelism for efficiency. Key optimizations include:

- **GPU Dimension Index Construction:** Parallel indexing of dataset dimensions using thread blocks.
- **Local Skylines and Reduction:** Computation of partial skyline sets per block, merged using a reduction tree.
- **Efficient Dominance Checks:** Optimized comparisons through warp-level processing, dimensional pruning, and shared memory utilization.
- **Global Skyline Consolidation:** Merging local skylines using atomic operations to ensure correctness.

**Skyline Maintenance with Incoming Data:** To maintain skyline correctness in dynamic data streams, *ECSQ* updates its dimension index efficiently. New tuples are inserted based on their preference order and evaluated against existing skyline points. The lower-bounded dimension  $A_{lower}$ , containing the fewest skyline tuples, is used for initial dominance checks:

$$A_{lower} = \arg \min_{A_i} \left| \frac{(v_x^i - \min(A_i))}{\max(A_i) - \min(A_i)} \right| \quad (1)$$

Similarly, the upper-bounded dimension  $A_{upper}$ , which determines potential skyline tuple removals, is computed as:

$$A_{upper} = \arg \max_{A_i} \left| \frac{(v_x^i - \min(A_i))}{\max(A_i) - \min(A_i)} \right| \quad (2)$$

By distributing these computations across GPU threads, *ECSQ* ensures efficient skyline maintenance with minimal synchronization overhead, making it suitable for real-time analytics.

### IV. GPU E-COMMERCE SKYLINE QUERY (ECSQ)

*ECSQ* is the GPU-accelerated version of PRSS [10] for E-Commerce datasets, using GPU parallelism for efficient skyline computation over E-Commerce data streams. It updates the skyline incrementally by processing incoming and expired tuples in parallel, optimizing memory access, and using warp-level synchronization.

#### GPU Kernel Workflow:

- 1) **Remove Expired Tuples:** Flags and removes expired tuples using parallel reduction.
- 2) **Parallel Dominance Checking:** Determines tuple dominance using warp-level primitives.
- 3) **Skyline Update:** Inserts new skyline tuples and removes dominated ones.

#### Data Structures:

- **Dimension Index ( $A$ ):** Stores tuples based on dimensions.
- **Skyline Index ( $S$ ):** Tracks current skyline tuples.

#### Algorithm Execution:

- Expired tuples are identified and removed.
- Dominance checks update the skyline set  $S$ .
- Incoming tuples undergo range search and skyline insertion.

#### Key GPU Kernels:

- **Kernel 1: RemoveExpiredTuples( $A, Z$ )**

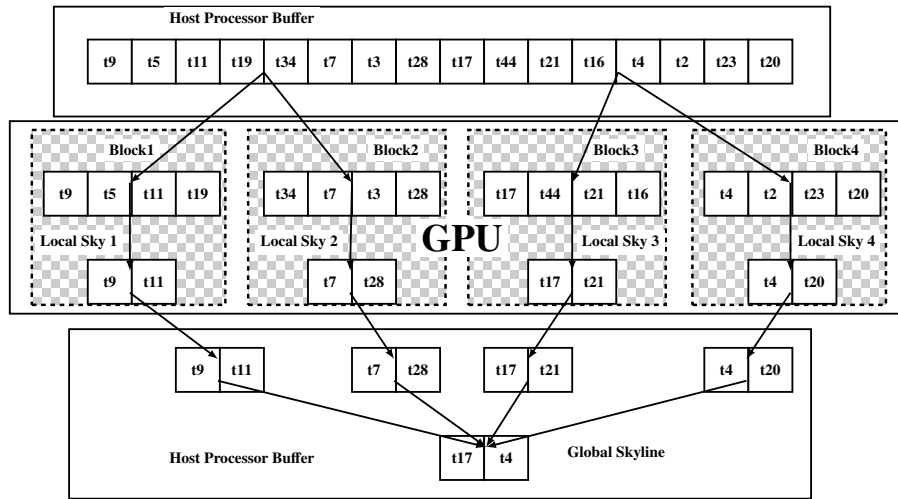


Fig. 3. Local Skyline computation on GPU blocks.

- **Kernel 2: CheckDominance( $A, x$ )**
- **Kernel 3: UpdateSkyline( $S, x$ )**

ECSQ efficiently updates skyline queries using GPU parallelism, reducing computational overhead compared to CPU-based approaches and making it suitable for real-time big data applications.

## V. PERFORMANCE EVALUATION

### A. Experimental Setup

We evaluate ECSQ using count-based sliding windows on synthetic and real-world data streams. The ECSQ implementation, written in CUDA and compiled with `nvcc v10.0.0`, runs on an NVIDIA GeForce RTX 4090 GPU. BNL [16], BskyTree [36], Salsa [50] and SFS [24] are implemented in C++ (compiled with `g++ v14.2.0, -O3` flag), using OpenMP API v5.0 for multi-threading. Experiments were conducted on an Intel Core i7 (4.2 GHz, 16 cores, 32 GB RAM) running Windows 10.

**Datasets:** We use synthetic and real-world datasets:

a) *Synthetic Data:* Generated using the Skyline benchmarking tool<sup>1</sup>, with dimensionality  $d$  from 2 to 24 and cardinality  $n$  from 100k to 1M records.

b) *Real-World Data:*

- **Amazon product Dataset<sup>2</sup>:** Contains product reviews and metadata from Amazon, including approximately millions of records across various categories. The data encompasses multiple attributes such as product descriptions, category information, price, brand, and image features.
- **Weather Dataset<sup>3</sup>:** Monthly precipitation data with coordinates and elevation for 566,268 global locations.
- **Covertypes Dataset**

<sup>4</sup>: This dataset contains cartographic variables such as elevation, proximity to the nearest road, and slope. The data is collected for 30m × 30m grid cells in the Roosevelt National Forest, Colorado, USA. Skyline points in this dataset correspond to forested areas with unique cartographic characteristics.

- **Weather Dataset**

<sup>5</sup>: This dataset provides monthly precipitation data along with geographical coordinates (latitude, longitude) and elevation for 566,268 terrestrial locations worldwide. Each record represents a 10-degree latitude-longitude grid cell. A skyline record in this dataset denotes a distinctive pattern of months with unusually high rainfall, favoring higher elevations and northeastern locations.

- **MovieLens Dataset: <sup>6</sup>** Contains 3.2 million ratings across 87,585 movies from 200,948 users. Includes movie metadata (title, genres) and user ratings (0.5-5 stars). Derived features include average rating, rating count, release year, and genre indicators (5-20 dimensions). Find Skyline movies that dominate in both popularity and rating quality. Identify niche films that are highly rated but less known. Optimize recommendation systems by finding undominated preference combinations.
- **OpenSense Air Quality Dataset: <sup>7</sup>** Contains > 200184 ultrafine particle (UFP) measurements and O<sub>3</sub>/CO readings from mobile sensors (2012-2014). 6-10 dimensions including pollutant concentrations, location, temperature, and humidity. Skyline Extraction Detect extreme pollution events (high values across multiple pollutants). Identify sensor locations with worst combined air quality metrics and Optimize sensor placement by finding coverage/accuracy trade-offs.

<sup>1</sup><http://pgfoundry.org/projects/randdataset>

<sup>2</sup><https://arxiv.org/html/2410.05763v3>

<sup>3</sup><https://crudata.uea.ac.uk/cru/data/hrg/tmc/>

<sup>4</sup><https://doi.org/10.24432/C50K5N>

<sup>5</sup><https://crudata.uea.ac.uk/cru/data/hrg/tmc/>

<sup>6</sup><https://grouplens.org/datasets/movielens/32m/>

<sup>7</sup>[https://gitlab.ethz.ch/tec/public/opensense/-/tree/master?ref\\_type=heads](https://gitlab.ethz.ch/tec/public/opensense/-/tree/master?ref_type=heads)

**Algorithm 1: ECSQ - GPU Range Search for Optimal E-Commerce Products**


---

**Input** : E-commerce product stream  $\mathcal{P}$ , Window  $\mathcal{W}$ , Product attributes  $\mathcal{A}$ , start, end  
**Output** Instant update of optimal products  $S$ .

---

```

1 Global Memory:
2   Attribute Index  $A = \emptyset$ 
3   Optimal Products Index  $S = \emptyset$  // Initializes empty global indices for product comparison.
4 while  $t = \text{start}$  and  $t < \text{end}$  do
5   Launch GPU Kernel: Process Expired Products
6   ECSQ_ExpiredProducts<<<  $\text{grid}, \text{block}$  >>> ( $A, \mathcal{W}, V$ ) // Parallel retrieval of expired products from sliding window.
7   foreach  $\text{product} \in V$  in parallel (CUDA threads) do
8     if  $\text{product} \in S$  then
9       Launch GPU Kernel: Dominance Checks
10      ECSQ_DominatedProducts<<<  $\text{grid}, \text{block}$  >>> ( $A, \text{product}, E$ ) // Finds products directly dominated by current product.
11      Disable  $\text{product} \in A$  // Removes product from active comparison set.
12      foreach  $\text{candidate} \in E$  in parallel do
13        if  $\text{GRangeSearch}(A, \text{candidate})$  then
14           $S \leftarrow S \cup \{\text{candidate}\}$  // Updates optimal products with new candidates.
15        end
16      end
17       $S \leftarrow S \setminus \{\text{product}\}$  // Removes expired product from optimal set.
18    end
19    Remove  $\text{product}$  from  $A$  // Permanently removes product from index.
20  end
21  Launch GPU Kernel: Range Search for New Products
22  ECSQ_RangeSearch<<<  $\text{grid}, \text{block}$  >>> ( $A, \text{new\_product}, R$ ) // Finds products directly dominated by new product.
23  foreach  $\text{dominated} \in R$  in parallel do
24    Remove  $\text{dominated}$  from  $A$  // Removes inferior products from consideration.
25  end
26   $S \leftarrow S \cup \{\text{new\_product}\}$  // Adds new optimal product to result set.
27  foreach  $\text{new\_product} \in \mathcal{A}$  in parallel do
28    Insert  $\text{new\_product}$  to  $A$  // Parallel insertion of new products into index.
29  end
30 end

```

---

**Algorithm 2: GPU RangeSearch for E-Commerce Product Selection**


---

**Input** : Attribute index  $A$ , product  $p$ , Product attributes  $\mathcal{A}$ .  
**Output** true if  $p$  is an optimal product.

---

```

1 Optimal Products Index  $S$ 
2 Launch GPU Kernel: Compute Lower-Bounded Attribute
3    $\text{lower} \leftarrow \text{ECSQ\_LowerBoundedAttribute}<<< \text{grid}, \text{block} >>> (A, p)$  // Parallel calculation of the most favorable attribute.
4 Launch GPU Kernel: Retrieve Product Blocks for Comparison
5    $B \leftarrow \text{ECSQ\_GetBlock}<<< \text{grid}, \text{block} >>> (p, A_{\text{lower}})$  // Parallel retrieval of similar products for comparison.
6 if not  $\text{ECSQ\_BNL}<<< \text{grid}, \text{block} >>> (B, p)$  then
7   return false // Stops if  $p$  is not locally optimal.
8 end
9 Launch GPU Kernel: Retrieve Lower-Bounded Optimal Products
10   $S \leftarrow \text{ECSQ\_LowerBoundedSkyline}<<< \text{grid}, \text{block} >>> (A_{\text{lower}}, p)$  // Parallel retrieval of current optimal products.
11 Parallel Dominance Check on Optimal Products
12 foreach  $\text{optimal} \in S$  in parallel do
13   if  $\text{optimal} < p$  then
14     return false // Stops if  $p$  is dominated by any optimal product.
15   end
16 end
17 Launch GPU Kernel: Compute Upper-Bounded Attribute
18    $\text{upper} \leftarrow \text{ECSQ\_UpperBoundedAttribute}<<< \text{grid}, \text{block} >>> (A, p)$  // Parallel calculation of least favorable attribute.
19 Launch GPU Kernel: Retrieve Product Blocks for Comparison
20    $B \leftarrow \text{ECSQ\_GetBlock}<<< \text{grid}, \text{block} >>> (p, A_{\text{upper}})$  // Parallel retrieval of products in worst attribute.
21 Parallel Dominance Check on Upper-Bounded Block
22 foreach  $\text{competitor} \in B$  in parallel do
23   if  $\text{competitor} \in S$  and  $p < \text{competitor}$  then
24      $\text{ECSQ\_UpdateDominanceList}<<< \text{grid}, \text{block} >>> (p, \text{competitor})$  // Parallel update of dominated products.
25      $S = S \setminus \{\text{competitor}\}$  // Removes inferior products from optimal set.
26   end
27 end
28 Launch GPU Kernel: Retrieve Upper-Bounded Optimal Products
29    $S \leftarrow \text{ECSQ\_UpperBoundedSkyline}<<< \text{grid}, \text{block} >>> (A_{\text{upper}}, p)$  // Parallel retrieval of products weak in key attributes.
30 Parallel Dominance Check on Upper-Bounded Products
31 foreach  $\text{suboptimal} \in S$  in parallel do
32   if  $p < \text{suboptimal}$  then
33      $\text{ECSQ\_UpdateDominanceList}<<< \text{grid}, \text{block} >>> (p, \text{suboptimal})$  // Parallel update of dominated products.
34      $S = S \setminus \{\text{suboptimal}\}$ 
35   end
36 end
37 return true // Returns true if  $p$  is an optimal product.

```

---

**Algorithm 3:** ECSQ\_Dominate( $Product_1, Product_2, \mathcal{A}$ )

---

**Input :**  $Product_1, Product_2$ , Product attributes  $\mathcal{A}$ .  
**Output** True or False.

---

```

1 procedure ECSQ_Dominate( $Product_1, Product_2, \mathcal{A}$ )
2    $flag \leftarrow 0$  // Initialize product superiority flag.
3   Launch GPU Kernel: Parallel Product Comparison
4   for each thread attribute in parallel (0 to  $\mathcal{A}-1$ ) do
5     if  $Product_1[attribute] > Product_2[attribute]$ 
6       then
7         return 0 // Stops if Product_1 is worse in any attribute (e.g., higher price).
8       end
9     else if  $Product_1[attribute] < Product_2[attribute]$ 
10      then
11         $flag \leftarrow 1$  // Marks that Product_1 is better in at least one attribute (e.g., lower price).
12      end
13    end
14  Synchronize Threads // Ensure all attribute comparisons complete.
15  return  $flag$  // Product_1 dominates Product_2 if flag remains 1.

```

---

**Algorithm 4:** Main GPU Parallel Optimal Product Selection

---

**Input :** Window  $\mathcal{W}$ , Product attributes  $\mathcal{A}$ ,  $datatype$ ,  $dataset\_size$ ,  $num\_blocks$ ,  $threads\_per\_block$   
**Output**  $globalOptimalProducts[ ]$

---

```

1  $product\_data \leftarrow readProductStream(dataset\_size, \mathcal{A}, datatype)$ 
2  $step \leftarrow dataset\_size / num\_blocks$ 
3  $localOptimal[ ]$ 
4 // Step 1: Compute Local Optimal Products in Parallel on GPU Launch Kernel:
  ECSQ_Kernel <<<
  num_blocks, threads_per_block >>>
  (product_data, step,  $\mathcal{A}$ , localOptimal)
5 // Step 2: Merge Local Optimal Products in Parallel
  num_active_blocks  $\leftarrow num\_blocks$ 
6 while  $num\_active\_blocks > 1$  do
7    $num\_active\_blocks \leftarrow num\_active\_blocks / 2$ 
8   Launch Kernel: Merge_Local_Optimal <<<
    num_active_blocks, threads_per_block >>>
    (localOptimal)
9 end
10 Copy  $localOptimal[0]$  to  $globalOptimalProducts[ ]$ 

```

---

- **Intel Berkeley Lab Sensor Data:**<sup>8</sup> Contains 2.3 million readings from 54 sensors monitoring temperature, humidity, light, and voltage (2004 deployment). 8 raw dimensions including temporal, environmental, and sensor status metrics. Using Skyline query to find optimal sensors (high voltage + stable readings), Detect anomalous measurements (extreme combinations of environmental factors) and Identify energy-efficient configurations (temperature-voltage trade-offs).

**B. Experimental Results**

We compare ECSQ against BNL [16], BskyTree [36], Salsa [50] and SFS [24] for different dataset cardinalities and dimensionalities across anticorrelated, correlated, and independent distributions.

**Impact of Cardinality (Fixed  $d = 16$ ):** Figures 4 show execution times across varying cardinalities. ECSQ significantly outperforms competitors, with performance gains widening as dataset size increases. ECSQ scales efficiently, maintaining consistent performance for datasets from  $10^3$  to  $10^6$  records. These results confirm ECSQ's superiority for large-scale skyline queries, making it a powerful solution for real-time analytics in big data applications.

**Impact of Dimensionality (Fixed Cardinality = 1M):** Figures 5 demonstrate ECSQ's superior scalability in multi-dimensional scenarios, mitigating the curse of dimensionality using GPU parallelism. Low-dimensional datasets (2-4D) show near-linear speedup, while multi-dimensional datasets (8-20D) benefit from shared memory and warp-level execution.

**Real-World Dataset Performance:****VI. CONCLUSION**

This paper presented an efficient GPU-based skyline algorithm optimized for real-time data streams using a count-based sliding window model. using the massive parallelism of GPUs, our approach significantly accelerates skyline computations compared to traditional single-core and multicore methods. Extensive experiments on synthetic and real-world datasets validated the algorithm's scalability and robustness across diverse data distributions. Our results consistently demonstrated superior performance over existing skyline algorithms, highlighting its efficiency in handling E-Commerce data and large-scale streams. The algorithm's responsiveness and computational efficiency make it well-suited for applications requiring real-time decision-making, such as monitoring systems and dynamic data analytics. The integration of advanced GPU memory management and parallelization techniques further enhances its adaptability to modern high-performance computing environments. Future research may explore optimizing its performance in heterogeneous architectures, extending support to categorical, incomplete, and uncertain data, and integrating machine learning techniques for predictive analytics. In summary, our GPU-based skyline algorithm represents a significant advancement in skyline computation, offering a scalable and efficient solution for real-time data processing in dynamic environments.

<sup>8</sup><https://www.kaggle.com/datasets/divyansh22/intel-berkeley-research-lab-sensor-data>

Dataset	Window Size	Dim.	BNL Time (s)	SFS Time (s)	SalSa Time (s)	BSkyTree Time (s)	ECSQ Time (s)
Amazon	581012	16	162.98	141.31	230.98	135.27	115.24
Weather	566262	15	254.62	221.24	340.56	220.38	117.23
MovieLens	3200000	5	5614.20	4812.68	1114.15	3825.31	825.32
OpenSense	200184	6	143.27	141.68	106.65	180.66	80.09
Intel Berkeley	2300000	4	3518.23	2482.56	1132.05	1133.11	815.46

TABLE II. Performance Comparison of Skyline Algorithms on Real-World Datasets

## REFERENCES

- [1] W. Choi, L. Liu, and B. Yu, "Multi-criteria decision making with skyline computation," in *2012 IEEE 13th International Conference on Information Reuse & Integration (IRI)*. IEEE, 2012, pp. 316–323.
- [2] B. Jiang and X. Du, "Personalized travel route recommendation with skyline query," in *2018 IEEE 9th International Conference on Dependable Systems, Services and Technologies (DESSERT)*. IEEE, 2018, pp. 549–554.
- [3] W. Khames, A. Hadjali, and M. Lagha, "Skyline computation on multicore architectures: A survey," in *2020 International Conference on Data Analytics for Business and Industry: Way Towards a Sustainable Economy (ICDABI)*. IEEE, 2020, pp. 1–6.
- [4] W. Dhifli, N. E. I. Karabadi, and M. Elati, "Evolutionary mining of skyline clusters of attributed graph data," *Information Sciences*, vol. 509, pp. 501–514, 2020.
- [5] R. Soundararajan, S. R. Kumar, N. Gayathri, and F. Al-Turjman, "Skyline query optimization for preferable product selection and recommendation system," *Wireless Personal Communications*, vol. 117, pp. 3091–3108, 2021.
- [6] W. Hanning, X. Weixiang, J. Yang, L. Wei, and J. Chaolong, "Efficient processing of continuous skyline query over smarter traffic data stream for cloud computing," *Discrete Dynamics in Nature and Society*, vol. 2013, 2013.
- [7] Y.-W. Peng and W.-M. Chen, "Parallel k-dominant skyline queries in high-dimensional datasets," *Information Sciences*, vol. 496, pp. 538–552, 2019.
- [8] R. Liu and D. Li, "Dynamic dimension indexing for efficient skyline maintenance on data streams," in *International Conference on Database Systems for Advanced Applications*. Springer, 2020, pp. 272–287.
- [9] M. Amiruzzaman and S. Jamonnak, "Multi-dimensional skyline query to find best shopping mall for customers," in *2020 6th Conference on Data Science and Machine Learning Applications (CDMA)*. IEEE, 2020, pp. 71–76.
- [10] W. Khames, A. Hadjali, and M. Lagha, "Parallel continuous skyline query over high-dimensional data stream windows," *Distributed and Parallel Databases*, pp. 1–56, 2024.
- [11] K. S. Bøgh, I. Assent, and M. Magnani, "Efficient gpu-based skyline computation," in *Proceedings of the Ninth International Workshop on Data Management on New Hardware*, 2013, pp. 1–6.
- [12] K. S. Bøgh, S. Chester, and I. Assent, "Skylalign: a portable, work-efficient skyline algorithm for multicore and gpu architectures," *The VLDB Journal*, vol. 25, no. 6, pp. 817–841, 2016.
- [13] C. Li, Y. Gu, J. Qi, and G. Yu, "Parallel skyline processing using space pruning on gpu," in *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*, 2022, pp. 1074–1083.
- [14] W.-M. Chen, H.-H. Tsai, and J. F. Ling, "Parallel computation of dominance scores for multidimensional datasets on gpus," *IEEE Transactions on Parallel and Distributed Systems*, 2024.
- [15] K. S. Bøgh, S. Chester, D. Šidlauskas, and I. Assent, "Template skycube algorithms for heterogeneous parallelism on multicore and gpu architectures," in *Proceedings of the 2017 ACM International Conference on Management of Data*, 2017, pp. 447–462.
- [16] S. Borzsony, D. Kossmann, and K. Stocker, "The skyline operator," in *Proceedings 17th international conference on data engineering*. IEEE, 2001, pp. 421–430.
- [17] H. Lu, Y. Zhou, and J. Haustad, "Continuous skyline monitoring over distributed data streams," in *Scientific and Statistical Database Management: 22nd International Conference, SSDBM 2010, Heidelberg, Germany, June 30–July 2, 2010. Proceedings 22*. Springer, 2010, pp. 565–583.
- [18] I. Kertiou, S. Benharzallah, L. Kahloul, M. Beggas, R. Euler, A. Laouid, and A. Bounceur, "A dynamic skyline technique for a context-aware selection of the best sensors in an iot architecture," *Ad Hoc Networks*, vol. 81, pp. 183–196, 2018.
- [19] A. Das Sarma, A. Lall, D. Nanongkai, and J. Xu, "Randomized multi-pass streaming skyline algorithms," *Proceedings of the VLDB Endowment*, vol. 2, no. 1, pp. 85–96, 2009.
- [20] F. Guo, Y. Mai, J. Tang, Y. Huang, and L. Zhu, "Robust and automatic skyline detection algorithm based on mssdn," *Journal of Advanced Computational Intelligence and Intelligent Informatics*, vol. 24, no. 6, pp. 750–762, 2020.
- [21] K. Alami and S. Maabout, "A framework for multidimensional skyline queries over streaming data," *Data & Knowledge Engineering*, vol. 127, p. 101792, 2020.
- [22] A. N. Papadopoulos, E. Tiakas, T. Tzouramanis, N. Georgiadis, and Y. Manolopoulos, "Skylines and other dominance-based queries," *Synthesis Lectures on Data Management*, vol. 15, no. 2, pp. 1–158, 2020.
- [23] H. Im and S. Park, "Group skyline computation," *Information Sciences*, vol. 188, pp. 151–169, 2012.
- [24] C. Sheng and Y. Tao, "On finding skylines in external memory," in *Proceedings of the thirtieth ACM SIGMOD-SIGACT-SIGART symposium on principles of database systems*, 2011, pp. 107–116.
- [25] M. E. Khalefa, M. F. Mokbel, and J. J. Levandoski, "Skyline query processing for incomplete data," in *2008 IEEE 24th international conference on data engineering*. IEEE, 2008, pp. 556–565.
- [26] Y. Gao, X. Miao, H. Cui, G. Chen, and Q. Li, "Processing k-skyband, constrained skyline, and group-by skyline queries on incomplete data," *Expert Systems with Applications*, vol. 41, no. 10, pp. 4959–4974, 2014.
- [27] S. Zhang, N. Mamoulis, and D. W. Cheung, "Scalable skyline computation using object-based space partitioning," in *Proceedings of the 2009 ACM SIGMOD International Conference on Management of data*, 2009, pp. 483–494.
- [28] M. Endres and E. Glaser, "Indexing for skyline computation: A comparison study," in *Flexible Query Answering Systems: 13th International Conference, FQAS 2019, Amantea, Italy, July 2–5, 2019, Proceedings 13*. Springer, 2019, pp. 31–42.
- [29] D. Papadias, Y. Tao, G. Fu, and B. Seeger, "An optimal and progressive algorithm for skyline queries," in *Proceedings of the 2003 ACM SIGMOD international conference on Management of data*, 2003, pp. 467–478.
- [30] S. Sun, Z. Huang, H. Zhong, D. Dai, H. Liu, and J. Li, "Efficient monitoring of skyline queries over distributed data streams," *Knowledge and information systems*, vol. 25, no. 3, pp. 575–606, 2010.
- [31] W.-T. Balke, U. Güntzer, and J. X. Zheng, "Efficient distributed skylining for web information systems," in *Advances in Database Technology-EDBT 2004: 9th International Conference on Extending Database Technology, Heraklion, Crete, Greece, March 14–18, 2004 9*. Springer, 2004, pp. 256–273.
- [32] E. Lo, K. Y. Yip, K.-I. Lin, and D. W. Cheung, "Progressive skylining over web-accessible databases," *Data & Knowledge Engineering*, vol. 57, no. 2, pp. 122–147, 2006.
- [33] K.-L. Tan, P.-K. Eng, B. C. Ooi *et al.*, "Efficient progressive skyline computation," in *VLDB*, vol. 1, 2001, pp. 301–310.
- [34] K. C. Lee, W.-C. Lee, B. Zheng, H. Li, and Y. Tian, "Z-sky: an efficient skyline query processing framework based on z-order," *The VLDB Journal*, vol. 19, pp. 333–362, 2010.
- [35] J. Selke and W.-T. Balke, "Skymap: a trie-based index structure for high-performance skyline query processing," in *International Conference on Database and Expert Systems Applications*. Springer, 2011, pp. 350–365.
- [36] J. Lee and S.-w. Hwang, "Bskytree: scalable skyline computation using a balanced pivot selection," in *Proceedings of the 13th International Conference on Extending Database Technology*, 2010, pp. 195–206.
- [37] K. Koizumi, P. Eades, K. Hiraki, and M. Inaba, "Bjr-tree: fast skyline computation algorithm for serendipitous searching problems," in *2017*

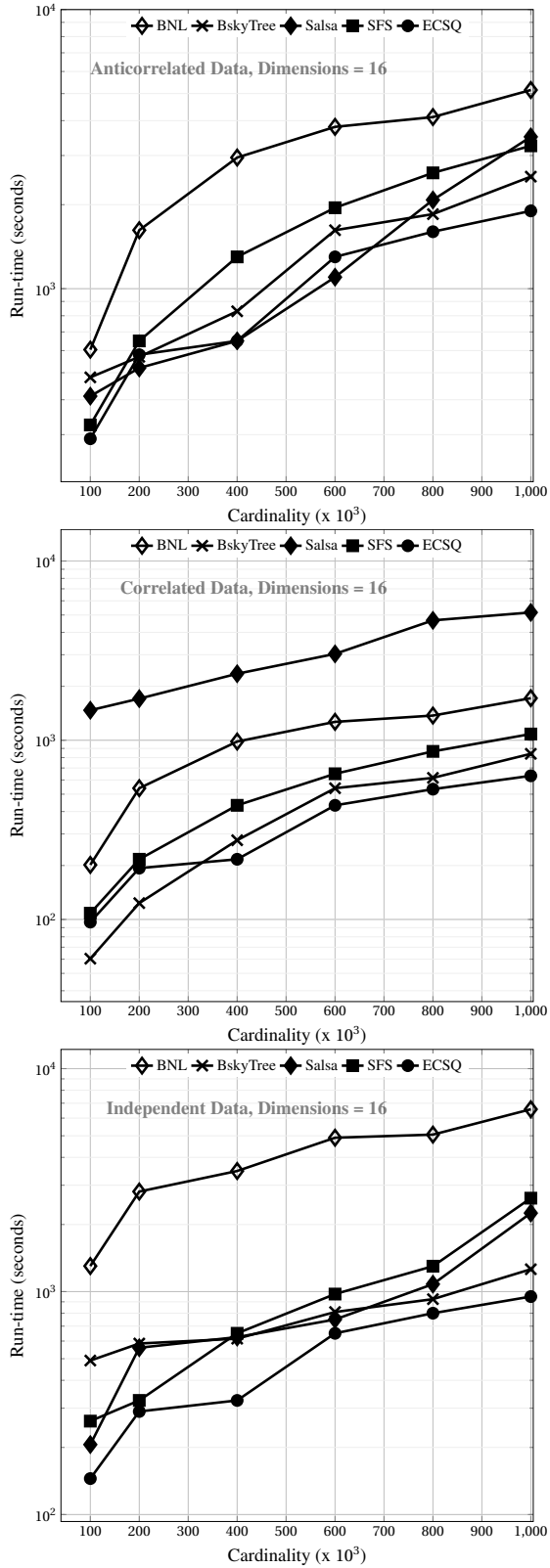


Fig. 4. Execution times for varying dataset cardinalities.

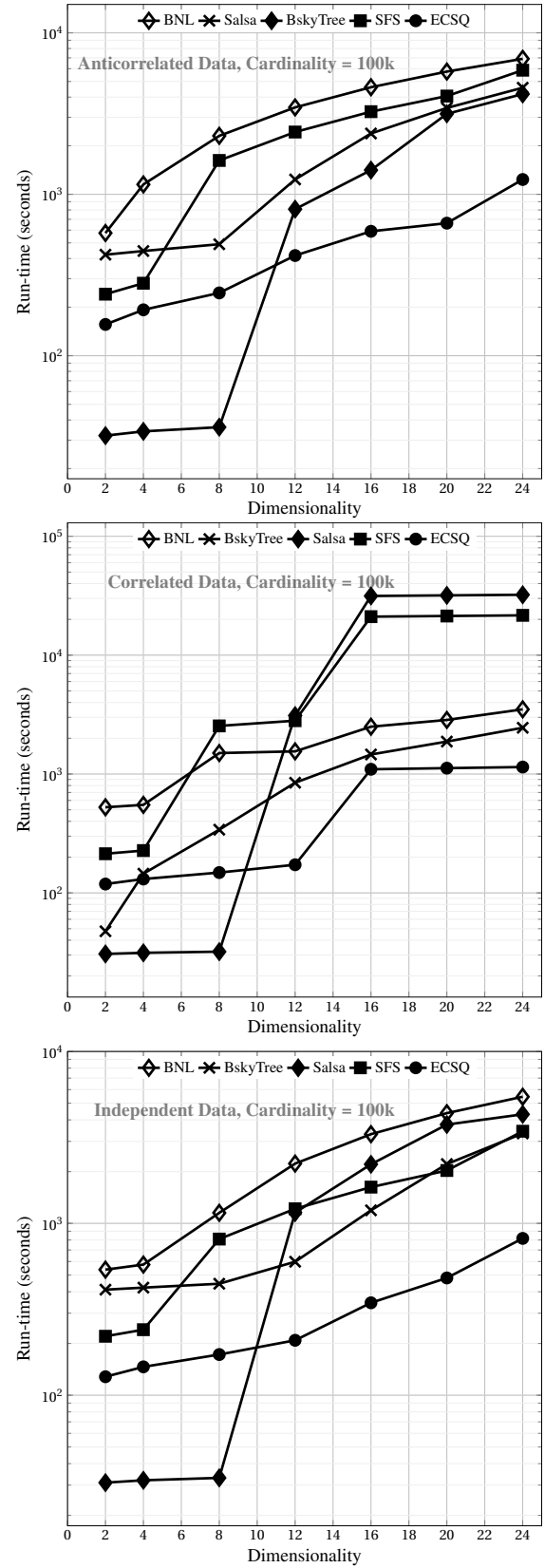


Fig. 5. Execution times for varying dataset dimensionalities.

- IEEE International Conference on Data Science and Advanced Analytics (DSAA)*. IEEE, 2017, pp. 272–282.
- [38] W.-T. Balke and U. Güntzer, “Supporting skyline queries on categorical data in web information systems,” in *RIDE-DM'04: Proceedings of the 14th International Workshop on Research Issues on Data Engineering: Web Services for E-Commerce and E-Government Applications*. IEEE, 2004, pp. 1–8.
  - [39] Q. Feng, Y. Dai, and K. Hwang, “Rainbow product ranking for upgrading e-commerce,” *IEEE Internet Computing*, vol. 13, no. 5, pp. 72–80, 2009.
  - [40] C. Lofi, U. Güntzer, and W.-T. Balke, “Eliciting skyline trade-offs using example-based heuristics for e-commerce applications,” in *2010 IEEE International Conference on e-Business Engineering*. IEEE, 2010, pp. 1–8.
  - [41] C. Lofi, W.-T. Balke, and U. Güntzer, “Eliciting customer wishes using example-based heuristics in e-commerce applications,” in *2011 IEEE 13th Conference on Commerce and Enterprise Computing*. IEEE, 2011, pp. 1–8.
  - [42] Y.-C. Chung, I.-F. Su, and C. Lee, “Efficient computation of combinatorial skyline queries,” *Information Systems*, vol. 38, no. 3, pp. 369–387, 2013.
  - [43] M. Sessoms and K. Anyanwu, “Skypackage: From finding items to finding a skyline of packages on the semantic web,” in *Joint International Semantic Technology Conference*, ser. LNCS 7774. Springer, 2013, pp. 49–64.
  - [44] A. Das Sarma, N. Parikh, and N. Sundaresan, “E-commerce product search: Personalization, diversification, and beyond,” in *Proceedings of the 23rd ACM International Conference on Conference on Information and Knowledge Management*. ACM, 2014, pp. 2009–2010.
  - [45] X. Zhou, K. Li, G. Xiao, Y. Zhou, and K. Li, “Top k favorite probabilistic products queries,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 11, pp. 3080–3093, 2016.
  - [46] X. Zhou, K. Li, Z. Yang, and K. Li, “Finding optimal skyline product combinations under price promotion,” *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 4, pp. 641–654, 2018.
  - [47] M. S. Islam, W. Rahayu, C. Liu, T. Anwar, and B. Stantic, “Computing influence of a product through uncertain reverse skyline,” in *Proceedings of the 29th International Conference on Scientific and Statistical Database Management*, 2017, pp. 1–12.
  - [48] B. Yin, K. Gu, X. Wei, S. Zhou, and Y. Liu, “A cost-efficient framework for finding prospective customers based on reverse skyline queries,” *Knowledge-Based Systems*, vol. 152, pp. 117–135, 2018.
  - [49] L. K. Tai, E. T. Wang, and A. L. Chen, “Finding the most profitable candidate product by dynamic skyline and parallel processing,” *Distributed and Parallel Databases*, vol. 39, pp. 979–1008, 2021.
  - [50] I. Bartolini, P. Ciaccia, and M. Patella, “Salsa: Computing the skyline without scanning the whole sky,” in *Proceedings of the 15th ACM international conference on Information and knowledge management*, 2006, pp. 405–414.

# Mining sequential patterns with quantities under constraints

1<sup>st</sup> Kemmar Amina

Oran Graduate School of Economics  
LITIO, University of Oran 1 - Ahmed Ben Bella  
Oran, Algeria  
amina.kemmar@ese-oran.dz

2<sup>nd</sup> Djebbar Amel Mounia

Oran Graduate School of Economics  
Oran, Algeria  
amel.djebbar@ese-oran.dz

3<sup>rd</sup> Touati Chahira

LITIO, University of Oran 1 - Ahmed Ben Bella  
Oran, Algeria  
touati.chahira@univ-oran1.dz

4<sup>th</sup> Kemmar Omar

University of Relizane - Ahmed Zabana  
Relizane, Algeria  
omar.kemmar@univ-relizane.dz

**Abstract**—This research focuses on the extraction of sequential patterns from datasets composed of sequences, specifically addressing the scenario where each item within a sequence is characterized by an associated quantity. Several methods were proposed to handle this problem without considering item quantities but only a few consider this scenario. To the best of our knowledge, there is only one CP based method allowing to handle this kind of sequences under constraints proposed in [8]. Constraint programming (CP) has demonstrated its effectiveness for solving sequential pattern mining problems. However, in this paper, we want to emphasize the flexibility offered by constraint programming, which allows us to express a variety of constraints in a very simple way without worrying about the implementation that has already been done. The constraints to be modeled are size constraint, membership constraint and regular expression constraint. We intend to soon experiment these constraints on different sequence datasets to show their value in practical applications.

**Index Terms**—sequence datasets, sequential patterns, quantitative items, constraints, constraint programming.

## I. INTRODUCTION

The substantial increase in data volume within various sectors mandates the application of data mining for effective analytical purposes. This paper explores the sequential pattern mining problem (SPM), a method used to extract meaningful patterns from sequence datasets. SPM finds applications in healthcare, education, web usage, bioinformatics, and telecommunications. A prominent application is

market basket analysis, where customer purchase sequences are analyzed. These sequences consist of ordered itemsets, representing items bought concurrently. For example,  $\langle bread(2)eggs(10)chese(2) \rangle$  depicts a customer's sequential purchase, including quantities. Let's take another example by imagining a patient which is hospitalized and their vital signs are recorded multiple times a day. We can represent this data as sequences, where each sequence corresponds to a day of hospitalization  $\langle Temperature(38.5), Pulse(90), BloodPressure(140/90), Oxygen(95) \rangle$  which means that the patient has a temperature of 38.5° C, a pulse of 90 beats per minute, a blood pressure of 140/90 mmHg, and an oxygen saturation of 95%. Integrating quantities into the sequence allows for a better representation of the data, which makes it possible to extract more relevant patterns based on this new information.

Several methods have been proposed to solve the problem of extracting frequent patterns from a sequence database. These can be classified into two categories: classical methods and constraint programming based methods. To the best of our knowledge, there exists only one library Seq2Pat [19] which allows to impose constraints on quantified sequence datasets. However, the proposed method is not scalable, hence the need for another method that can handle large sequence databases with quantities.

In [8], we proposed the first CP approach to handle

sequences with quantities. In this paper, we want to emphasize the flexibility offered by constraint programming, which allows us to express a variety of constraints in a very simple way without worrying about the implementation that has already been done. We intend to soon test these constraints on different sequence datasets to show their value in practical applications.

The paper is organized as follows. Section II recalls preliminaries to understand the QSPM problem and constraint programming. Section III offers an overview of classical methods and CP approaches for sequential pattern mining under constraints. In section IV, we detail our global constraint by describing its filtering algorithm based on the principle of projected databases. Section V presents the CSP modeling for several constraints. Finally, we conclude and draw some perspectives in Section VI.

## II. PRELIMINARIES AND PROBLEM STATEMENT

First, we provide the basic definitions for sequential pattern mining in the context of sequences of items with quantities. Then, we present the concept of projected-databases, introduced in PrefixSpan algorithm [13]. Finally, we give an overview of constraint programming.

### A. Fundamentals of Sequential Pattern Mining

Let  $\mathcal{I}$  be a finite set of distinct *items* and  $Q$  a finite set of quantities. We call a quantitative item, each item  $i$  with its quantity  $q$  denoted as  $i(q)$ , where  $i \in \mathcal{I}, q \in Q$ . The quantity values of an item  $i$  are represented by  $Q_i = \{q_{i1} \dots q_{ik}\}$  in an increasing order. A quantitative sequence  $s$ , denoted q-sequence, is an ordered list  $\langle i_1(q_1) i_2(q_2) \dots i_n(q_n) \rangle$ , where  $i_j(q_j)$ ,  $1 \leq j \leq n$ , is an extended item with its quantity.  $n$  is called the length of the q-sequence  $s$ . A quantitative sequence database  $QDB$  is a set of tuples  $(sid, s)$ , where  $sid$  is a sequence identifier and  $s$  a q-sequence denoted by  $QDB[sid]$ . Mining sequential patterns is based on the subsequence relation which is defined below taking into account the quantities of items:

**Definition 1 (subsequence relation):** A q-sequence  $\alpha = \langle \alpha_1(q'_1) \dots \alpha_m(q'_m) \rangle$  is a subsequence of  $s = \langle i_1(q_1) i_2(q_2) \dots i_n(q_n) \rangle$ , denoted by  $(\alpha \preceq s)$ , if  $m \leq n$  and there exist integers  $1 \leq j_1 \leq \dots \leq j_m \leq n$ , such that  $(\alpha_j = i_{j_j})$  and  $(q'_j \leq q_{j_j})$  for all  $1 \leq j \leq m$ . We also say that  $\alpha$  is contained in  $s$  or  $s$  is a super-sequence of  $\alpha$ . A tuple  $(sid, s)$  contains a q-sequence  $\alpha$ , if  $\alpha \preceq s$ .

For example, the sequence  $\langle b(4)a(1) \rangle$  is a sub-sequence of  $\langle c(2)b(5)c(3)a(1) \rangle$  but not of  $\langle b(2)a(1)c(3) \rangle$ .

The subsequence relation allows to define the cover of a q-sequence  $\alpha$  which consists in all tuples in  $QDB$  containing  $\alpha$ . Thus, the support of  $\alpha$  is the cardinal of its cover:  $\text{sup}(\alpha)_{QDB} = |\{(sid, s) \in QDB \mid \alpha \preceq s\}|$ . We can define now the problem of mining quantitative sequential patterns as follows:

**Definition 2 (Quantitative sequential pattern mining (QSPM)):** Given a quantitative sequence database  $QDB$  and a minimum support threshold  $\text{minsup}$ . The problem of quantitative sequential pattern mining (QSPM) is to find all q-patterns  $p$  such that  $\text{sup}_{QDB}(p) \geq \text{minsup}$ .

Customer Id	Customer Sequence
1	<sweet(4) pant(1) socks(3)>
2	<sweet(2) pant(2) scarf(2)>
3	<sweet(3) pant(2) socks(2) >

TABLE I  
CUSTOMER PURCHASES REPRESENTED AS SEQUENCES WITH QUANTITIES

Instead of considering only the minimum support constraint, this paper address the problem of extraction patterns verifying other interesting constraints like size, membership and regular expression constraints.

**Example 1:** let us consider the market basket analysis application. In this one, we consider the purchases made by customers in a retail store. Each sequence represents the items purchased by a customer at different times (the different items are ordered following their purchase time). The set of all sequences forms a sequence database given in Table I. After a pre-processing step, this dataset is represented by the sequence database given in Table II.  $QDB_1$  contains three sequences where the set of items is  $\mathcal{I} = \{a, b, c, d\}$  and the set of all quantities is  $Q = \{1, 2, 3, 4, 6\}$ . The allowed quantities for each item are :  $Q_a = \{1, 3\}, Q_b = \{2, 3, 4, 6\}, Q_c = \{2, 3\}, Q_d = \{2\}$ . The sequence  $s = \langle a(1)c(2) \rangle$  has 2 quantitative items, we say that  $s$  is 2-length sequence. The q-pattern  $p = \langle a(1)c(1) \rangle$  is a subsequence of  $s$ :  $p \preceq s$ . If we consider  $\text{minsup} = 2$ , 14 q-sequences are extracted, the result of the mining process with details is given in TableIII.

In this paper, we address the problem of mining quantitative sequential patterns under constraints using constraint programming (CP). Differently to ad-hoc methods, CP offers an easy way to the user to express many constraints in a declarative way without considering new implementations. In the next section, we give an overview of CP necessary to understand our CP-approach.

sid	quantitative Sequence
1	$\langle b(4)a(1)c(3) \rangle$
2	$\langle b(2)a(2)b(6)d(2) \rangle$
3	$\langle b(3)a(2)c(2) \rangle$

TABLE II  
A QUANTITATIVE SEQUENCE DATABASE EXAMPLE  $QDB_1$ .

q-sequence	cover	support
$\langle b(2) \rangle, \langle b(2)a(1) \rangle, \langle b(3) \rangle, \langle a(1) \rangle$	$(1, s_1), (2, s_2), (3, s_3)$	3
$\langle b(2)a(1)c(2) \rangle, \langle c(2) \rangle, \langle a(1)c(2) \rangle, \langle b(2)c(2) \rangle, \langle b(3)c(2) \rangle, \langle b(3)a(1) \rangle$	$(1, s_1), (3, s_3)$	2
$\langle b(2)a(2) \rangle, \langle a(2) \rangle$	$(2, s_2), (3, s_3)$	2
$\langle b(4) \rangle, \langle a(1) \rangle$	$(1, s_1), (2, s_2)$	2

TABLE III  
QUANTITATIVE SEQUENTIAL PATTERNS EXTRACTED FROM  $QDB_1$  WHEN  $minsup = 2$ .

### B. Projected databases concept

We now present the necessary definitions to address the concept of projected databases introduced in [13] considering sequence databases without quantities.

**Definition 3 (prefix, projection, suffix):** Let  $\beta = \langle \beta_1 \dots \beta_n \rangle$  and  $\alpha = \langle \alpha_1 \dots \alpha_m \rangle$  be two sequences, such that  $m \leq n$ .

- The sequence  $\alpha$  is called prefix of  $\beta$  iff  $\forall i \in [1..m], \alpha_i = \beta_i$ .
- The sequence  $\beta = \langle \beta_1 \dots \beta_n \rangle$  is called a projection of a some sequence  $s$  w.r.t.  $\alpha$ , iff (1)  $\beta$  inf  $s$ , (2)  $\alpha$  is a prefix of  $\beta$  et (3) there is no proper super-sequence  $\beta'$  of  $\beta$  such that  $\beta'$  inf  $s$  and  $\beta'$  accepts  $\alpha$  as a prefix.
- The sequence  $\gamma = \langle \beta_{m+1} \dots \beta_n \rangle$  is called the suffix of  $s$  w.r.t.  $\alpha$ . Using the standard definition of the concatenation operator "concat", we have  $\beta = \text{concat}(\alpha, \gamma)$ .

**Example 2:** Let us consider the second sequence of  $QDB_1$  without quantities,  $s_2 = \langle b, a, b, d \rangle$ . For instance, the sequence  $\alpha = \langle ab \rangle$  is a prefix of  $\beta = \langle abd \rangle$  and  $\gamma = \langle d \rangle$  is the suffix of  $s_1$  w.r.t.  $\alpha$ .

The sequence  $\beta = \langle bd \rangle$  is the projection of  $s_2$  w.r.t.  $\alpha = \langle ba \rangle$ .

**Definition 4 (Projected database):** Let  $SDB$  be a sequence database. The  $\alpha$ -projection of the database, denoted  $SDB|_\alpha$ , contains the set of suffixes of the sequences in  $SDB$  w.r.t. prefix  $\alpha$ .

[13] proposed an efficient algorithm, called PrefixSpan, for the sequential pattern mining based on the principle of

prefix	projected database	the set of frequent items
$\langle a \rangle$	$(1, \langle c \rangle)$ $(2, \langle bd \rangle)$ $(2, \langle c \rangle)$	$\{c\}$
$\langle b \rangle$	$(1, \langle ac \rangle)$ $(2, \langle abd \rangle)$ $(2, \langle ac \rangle)$	$\{a, c\}$
$\langle c \rangle$	$(1, \langle \rangle)$  $(3, \langle ac \rangle)$	$\{\}$
$\langle ac \rangle$	$(1, \langle \rangle)$  $(3, \langle \rangle)$	$\{\}$
$\langle ba \rangle$	$(1, \langle c \rangle)$ $(2, \langle bd \rangle)$ $(3, \langle c \rangle)$	$\{c\}$
$\langle bc \rangle$	$(1, \langle \rangle)$  $(3, \langle \rangle)$	$\{\}$
$\langle bac \rangle$	$(1, \langle \rangle)$  $(3, \langle \rangle)$	$\{\}$

TABLE IV  
PROJECTED DATABASES GENERATED BY PREFIXSPAN ( $minsup = 2$ ).

*projected databases.* The method operates by partitioning the original sequence database into smaller, projected databases corresponding to the different patterns identified during extraction; solely their suffixes are preserved. Then, the subsequent sequential patterns are extracted from each projected database, utilizing only the locally frequent items.

**Example 3:** Let us take the sequence database given in Table II with  $minsup = 2$  (without quantities).

PrefixSpan begins with the analysis of database  $QDB_1$  to find the set of frequent items which results on  $\{a, b, c\}$ . Then, each one is used as a prefix for calculating projected databases from  $QDB_1$ . This process of database projection terminates when no frequent items is detected i.e. more sequential super-patterns can be generated. The mining process is illustrated in Table IV.

The Proposition 1 This defines the method for computing the support of sequence  $\gamma$  within  $SDB|_\alpha$  [13].

**Proposition 1 (Support Calculation):**

Given any sequence  $\gamma$  within the sequence database  $SDB$ , where  $\gamma$  is formed by the concatenation of a prefix  $\alpha$  and a suffix  $\beta$  (i.e.,  $\gamma = \text{concat}(\alpha, \beta)$ ), the support of  $\gamma$  in  $SDB$  is equal to the support of  $\beta$  in the projection of  $SDB$  onto  $\alpha$  (i.e.,  $sup_{SDB}(\gamma) = sup_{SDB|_\alpha}(\beta)$ ).

This proposition ensures that only the sequences in  $SDB$

obtained from  $\alpha$  are to be considered for calculating the support of the sequence  $\gamma$ . Furthermore, only suffixes with  $\alpha$  as a prefix need to be computed.

### C. Constraint programming Concepts

**Constraint programming (CP).** Constraint programming [17] is a powerful paradigm for solving combinatorial search problems modeled as constraints. It is based on the following principle: (1) the user specifies the problem in a declarative way as a constraint satisfaction problem (CSP); (2) the solver looks for the complete and correct set of solutions to the problem. In this way, the problem specification is separated from the search strategy.

**Constraint Satisfaction Problem (CSP).** A CSP consists of a set  $\mathcal{X}$  of  $n$  variables, a domain  $\mathcal{D}$  mapping each variable  $X_i \in \mathcal{X}$  to a finite set of values  $\mathcal{D}(X_i)$ , and a set of constraints  $\mathcal{C}$ . An assignment  $\sigma$  is a mapping from variables in  $\mathcal{X}$  to values in their domains. A constraint  $C \in \mathcal{C}$  is a subset of the cartesian product of the domains of the variables that occur in  $C$ . The goal is to find an assignment such that all constraints are satisfied.

*Example 4:* Let be the following CSP:

$$\begin{cases} \mathcal{X} = \{X_1, X_2, X_3\} \\ \mathcal{D}(X_1) = \mathcal{D}(X_2) = \mathcal{D}(X_3) = \{1, 2, 3\} \\ \mathcal{C} = \{C_1(X_1, X_2), C_2(X_1, X_3), C_3(X_2, X_3)\}, \text{ where,} \\ C_1(X_1, X_2) \equiv (X_1 \neq X_2) \\ C_1(X_1, X_3) \equiv (X_1 \neq X_3) \\ C_1(X_2, X_3) \equiv (X_2 \neq X_3) \end{cases}$$

The above CSP admits three solutions:  $(X_1 = 1, X_2 = 2, X_3 = 3)$ ,  $(X_1 = 3, X_2 = 1, X_3 = 2)$  and  $(X_1 = 2, X_2 = 3, X_3 = 1)$ .

In CP, the resolution process consists in combining iteratively search and propagation phases. The search phase consists in enumerating all possible partial instantiations of variables until finding a solution or proving that no solution exists. The constraint propagation phase allows to reduce search space by filtering values from variable domains which can not participate in any solution for the CSP. Thus, each constraint is associated with a propagator (i.e. a filtering algorithm) for deleting all values from variable domains which do not satisfy this constraint. Since a variable can participate in several constraints, modifications on domains are propagated by activating the propagators of these constraints. This propagation process is repeated on all constraints until no filtering is possible or a variable domain becomes empty.

**Global constraints.** provide shorthands to often-used combinatorial substructures. We present three global constraints. (1) Let  $X = \langle X_1, X_2, \dots, X_n \rangle$  be a sequence of  $n$  variables. Let  $V$  be a set of values,  $l$  and  $u$  be two integers s.t.  $0 \leq l \leq u \leq n$ , the constraint  $\text{Among}(X, V, l, u)$  states that each value  $a \in V$  should occur at least  $l$  times and at most  $u$  times in  $X$  [3]. (2) Given a deterministic finite automaton  $A$ , the constraint  $\text{Regular}(X, A)$  ensures that the sequence  $X$  is accepted by  $A$  [15]. (3) The CSP given in Example 4 can be modeled using the  $\text{AllDifferent}$  global constraint [16] as follows:  $\text{AllDifferent}(X_1, X_2, X_3)$ .

### III. RELATED WORKS

The problem of mining frequent patterns from a sequence database has been extensively studied, and numerous methods have been proposed. These methods can be divided into two categories which are presented below.

#### A. Standard methods

The SPM was first proposed in [1]. Since then, many efficient specialized approaches have been proposed: cSpade [21], SPIRIT [4], SMA [18], CloSpan [20] and Gap-BIDE [9] which are both extensions of PrefixSpan [14] to mine closed frequent patterns and closed frequent patterns with gap constraints respectively.

#### B. CP based methods

The methods we just mentioned are specialized methods, which require the revision of the entire source code when modifying or adding constraints. It should be noted that when the sequence database is large, the process of extracting all frequent patterns becomes increasingly complex. To address such a problem, the user can focus the search by incorporating constraints. However, other approaches based on constraint programming (CP) were proposed [10] [6] [12] [11]. CP offers a simple way to incorporate a variety of constraints imposed on the generated patterns.

In [5], the authors have proposed the global constraint PREFIX-PROJECTION which performs better comparing to the proposed methods. Aoga et al. [2] have further extended this work by combining ideas from pattern mining as well as from CP. They improve the efficiency of the previous global constraint using (i) last-position lists technique similar to the LAPIN algorithm [22] and (ii) ideas from trailing CP solvers to avoid unnecessary copying. These two works don't allow to directly handle gap constraints. Thus, in [7], the authors

proposed the global constraint GAP-SEQ enabling to handle the gap constraint combining with other types of constraints. Note that the cited CP approaches don't consider quantitative datasets. Recently, we proposed the global constraint Q-Prefix-Projection [8], which is an extension of the two approaches proposed in [5] and [7] in order to handle quantities.

The objective of this paper is to show how this approach [8] allows to specify many constraints, both on the structure of the extracted pattern or on the item quantities, like size, membership, regular expression and other constraints which can be also combined together.

#### IV. A GLOBAL CONSTRAINT FOR QSPM

In this section, we first present the CSP modeling considering quantities and then, we detail the filtering algorithm of the global constraint Q-Prefix-Projection.

##### A. A CSP modeling for QSPM: variable and domains

For the SPM problem without quantities, the pattern  $P$  of length  $\ell$  to be extracted is modeled with  $\ell$  variables  $\langle P_1, P_2, \dots, P_\ell \rangle$  s.t.  $\forall i \in [1 \dots \ell], D(P_i) = \mathcal{I} \cup \{\square\}$ . For QSPM problem, since each item has a quantity, the unknown quantitative pattern is modeled with  $2 \times \ell$  variables  $\langle P_1, P_2, P_3, P_4, \dots, P_{2 \times \ell} \rangle$ . For each item in position  $i$  (an even position), we associate its quantity in the next one  $i + 1$  (an odd position). Both items and quantities are encoded as integers.

The symbol  $\square$  ( $\square \notin \mathcal{I}$ ) stands for an empty item (or an empty quantity) and denotes the end of the sequence. Let  $FreqI$  and  $FreqQ$  be the set of frequent items and frequent quantities respectively in the initial database. The domains of variables are defined as follows:

- 1)  $D(P_1) = FreqI$  and  $D(P_2) = FreqQ$  to avoid the empty sequence,
- 2)  $\forall i \in \{3 \dots 2 \times \ell\}$ :
  - $D(P_i) = FreqI \cup \{\square\}$  if  $i$  is even,
  - $D(P_i) = FreqQ \cup \{\square\}$  if  $i$  is odd.

Constraint programming is based on domain filtering. In our case, there is some filtering rules allowing to filter considerably variable domains:

- When the length of the unknown pattern is  $k$  ( $k < \ell$ ), the last variables from the position  $2 \times k + 1$  are filled with the symbol  $\square$  as follows:  $\forall j \in [2 \times k + 1 \dots 2 \times \ell], (P_j = \square)$ .
- When an item variable is assigned to an empty symbol  $\square$ , its corresponding quantity variable is also assigned

to  $\square$ . Otherwise, if the item variable is not empty, then its corresponding quantity variable can not be empty. We obtain the following two rules ( $i$  corresponds to the position of an item variable):

- 1)  $P_i = \square \Rightarrow P_{i+1} = \square$ ,
- 2)  $P_i \neq \square \Rightarrow P_{i+1} \neq \square$ .

In the following, we give the definition of our global constraint called Q-PREFIX-PROJECTION, which is an extension of the global constraint proposed in [5] without considering gap constraint.

**Definition 5** (Q-PREFIX-PROJECTION global constraint): Let  $P = \langle P_1, P_2, \dots, P_{2 \times \ell} \rangle$  be a q-pattern of size  $\ell$ .  $\langle d_1, \dots, d_{2 \times \ell} \rangle \in D(P_1) \times \dots \times D(P_{2 \times \ell})$  is a solution of Q-PREFIX-PROJECTION  $(P, QDB, minsup)$  iff  $sup_{QDB}(\langle d_1(d_2)d_3(d_4) \dots d_{(2 \times \ell)-1}(d_{2 \times \ell}) \rangle) \geq minsup$ .

**Example 5:** Consider the sequence database of Table II with  $minsup = 2$  and  $\ell = 3$ . Let  $P = \langle P_1, P_2, P_3, P_4, P_5, P_6 \rangle$  with  $D(P_1) = \mathcal{I}$ ,  $D(P_2) = Q$ ,  $D(P_3) = D(P_5) = \mathcal{I} \cup \{\square\}$  and  $D(P_4) = D(P_6) = Q \cup \{\square\}$ . Suppose that  $\sigma(P_1) = b$ ,  $\sigma(P_2) = 3$ ,  $\sigma(P_3) = a$ ,  $\sigma(P_4) = 1$ ,  $\sigma(P_5) = c$  and  $\sigma(P_6) = 2$ . Since  $sup_{QDB}(\langle b(3)a(1)c(2) \rangle) = 2$ , the Q-PREFIX-PROJECTION holds.

##### B. The filtering algorithm for Q-PREFIX-PROJECTION global constraint

In Constraint Programming, the definition of a global constraint is based on its filtering algorithm: how to reduce the domains of variables during the enumeration process?

The pseudo-code of the filtering algorithm of Q-PREFIX-PROJECTION global constraint is detailed in [8]. In this paper, we give the general algorithm, in a simplified manner, allowing to reduce variable domains (see Algorithm 1).

#### V. MODELING CONSTRAINTS

In this section, we show how to model several constraints on the extracted patterns. We can distinguish two kinds of constraints:

- 1) Item constraints which are imposed on the pattern structure.
- 2) Quantities constraints which are imposed on the quantity variables.

**Size constraints.** Our CP modeling allow to control straightforwardly the minimum and maximum length of the sequence

**Algorithm 1: Q-PREFIX-PROJECTION**


---

**Data:**  $QDB$ : initial database;  $\sigma$ : current prefix  $\langle \sigma(P_1), \dots, \sigma(P_i) \rangle$ ;  
 $minsup$ : the minimum support threshold;  $FQ$ : locally frequent quantities;  $FI$ : locally Frequent items.

---

```

begin
1  Calculate the frequent items w.r.t  $minsup$  ( $FreqI$ ) ;
2  Calculate the frequent quantities ( $FreqQ$ ) ;
3  Initialize the first variable  $P_1$  with  $FreqI$  ;
4  initialize the second variable  $P_2$  with  $FreqQ$  ;
   while (there exists an instantiated variable  $P_i$ ) do
       if ( $\sigma(P_i) = \square$ ) then
           Instantiate the variables from  $P_{i+1}$  until  $P_{2 \times \ell}$  to  $\square$  ;
           return true ;
       else if ( $P_i$  corresponds to an item variable (i.e.  $i$  is odd)) then
           Filter the domain of the quantity variable  $P_{i+1}$  with Frequent
             quantities corresponding to the item  $P_i$  ;
5  Calculate the prefix  $\sigma$  ;
6  Calculate the prefix projection database  $QDB|_{\sigma}$  ;
   if ( $|QDB|_{\sigma}| < minsup$ ) then
       The current prefix cannot be extended ;
       return False ;
   else
7  Calculate frequent items  $FreqI$  in  $QDB|_{\sigma}$  ;
8  Calculate frequent quantities corresponding to the frequent
   items ;
9  Filter the domain of the next variable  $D(P_{i+1}) \leftarrow FreqI$  ;
10 Instantiate the next free variable  $i + 1$  ;

```

---

as well as the length of elements, using the predefined solver constraints.

$$minSize(P, \ell_{min}) \equiv \bigwedge_{i=1}^{i=2 \times \ell_{min}} (P_{i,1} \neq \square)$$

$$maxSize(P, \ell_{max}) \equiv \bigwedge_{i=2 \times \ell_{max} + 1}^{i=2 \times \ell} (P_{i,1} = \square)$$

**Membership constraint.** This constraint can be imposed either on the pattern or the quantities. We can use the among global constraints as follow :

Let  $V$  be a subset of items,  $l$  and  $u$  two integers s.t.  $0 \leq l \leq u \leq \ell$ .

$$item(P, V) \equiv \bigwedge_{v \in V} \text{Among}(P, \{v\}, l, u)$$

enforces that items of  $V$  should occur at least  $l$  times and at most  $u$  times in  $P$ . To forbid items of  $V$  to occur in  $P$ ,  $l$  and  $u$  must be set to 0.

Our CP modeling allows also to impose this constraint on some item of position  $i$  in the pattern with a specified value  $v$ , since the variable encoding allows to capture the position of each item in the sequential pattern:

$$item(P_i, v) \equiv \text{Among}(P_i, v, 1, 1)$$

**Regular expression constraint.**

In general, this constraint is imposed on the extracted pattern and not on the quantities. let  $A_{reg}$  be the deterministic finite automaton encoding the regular expression  $exp$ . Let  $P'$  be

the variables allowing to capture all item variables (i.e.  $P' = \langle P_1, P_3, P_5, \dots, P_{2 \times \ell - 1} \rangle$ ).

$$reg(P', exp) \equiv \text{Regular}(P', A_{reg})$$

To the best of our knowledge, the more efficient ad hoc method to handle regular expressions is SMA [18]. Unfortunately, this method doesn't allow such constraint on sequences with quantities.

**Other constraints on quantities.** We can impose different constraints on variable quantities using mathematical constraints. In order to give examples, we have to take a real database like datasets extracted from an e-commerce. We can analyze these kind of datasets by asking various questions for a given  $minsup$  value.

- 1) Is there customers who buy less than M products in the same time ?
- 2) Is there customers who buy only one item for each product ?
- 3) Is there customers who buy only 3 products with the quantities 1, 2, 3 respectively ?

In practice, CP enables us to easily answer these questions, as it allows constraints to be directly expressed and incorporated into the model without requiring additional implementations.

The above constraints can be expressed as follows:

- 1)  $const.1 \equiv \sum_{i=2}^{i=2 \times \ell} (i \% 2 = 0) (P_i < M)$
- 2)  $const.2 \equiv \bigwedge_{i=2}^{i=2 \times \ell} (i \% 2 = 0) (P_i = 1)$
- 3)

$$const.3 \equiv \left( \bigwedge_{i=7}^{i=2 \times \ell} P_i = \square \right) \wedge (P_2 = 1, P_4 = 2, P_6 = 3)$$

## VI. CONCLUSION

In this paper, we address the problem of mining quantitative sequential patterns. In the sequence database, each item is associated with its quantity. In [8], we proposed the first CP approach to handle sequences with quantities. In this paper, we shown the flexibility offered by constraint programming, which allows us to express a variety of constraints in a very simple way without worrying about the implementation that has already been done. We modeled different constraints like size, membership and regular expression constraints which can be also combined together. As future work, we intend to do an extensive experimentation on real databases to show the interest of this method comparing to existing ones.

## REFERENCES

- [1] R. Agrawal and R. Srikant. Mining sequential patterns. In Philip S. Yu and Arbee L. P. Chen, editors, *ICDE*, pages 3–14. IEEE Computer Society, 1995.
- [2] John O. R. Aoga, Tias Guns, and Pierre Schaus. An efficient algorithm for mining frequent sequence with constraint programming. In *Machine Learning and Knowledge Discovery in Databases - European Conference, ECML PKDD 2016, Riva del Garda, Italy, September 19-23, 2016, Proceedings, Part II*, pages 315–330, 2016.
- [3] N. Beldiceanu and E. Contejean. Introducing global constraints in CHIP. *Journal of Mathematical and Computer Modelling*, 20(12):97–123, 1994.
- [4] Minos N. Garofalakis, R. Rastogi, and K. Shim. Mining sequential patterns with regular expression constraints. *IEEE Trans. Knowl. Data Eng.*, 14(3):530–552, 2002.
- [5] A. Kemmar, S. Loudni, Y. Lebbah, P. Boizumault, and T. Charnois. PREFIX-PROJECTION global constraint for sequential pattern mining. In *Principles and Practice of Constraint Programming - 21st International Conference, CP 2015, Cork, Ireland, August 31 - September 4, 2015, Proceedings*, pages 226–243, 2015.
- [6] A. Kemmar, W. Ugarte, S. Loudni, T. Charnois, Yahia Lebbah, P. Boizumault, and B. Crémilleux. Mining relevant sequence patterns with cp-based framework. In *ICTAI*, pages 552–559, 2014.
- [7] Amina Kemmar, Samir Loudni, Yahia Lebbah, Patrice Boizumault, and Thierry Charnois. A global constraint for mining sequential patterns with GAP constraint. In *Integration of AI and OR Techniques in Constraint Programming - 13th International Conference, CPAIOR 2016, Banff, AB, Canada, May 29 - June 1, 2016, Proceedings*, pages 198–215, 2016.
- [8] Amina Kemmar, Chahira Touati, and Yahia Lebbah. A cp-based approach for mining sequential patterns with quantities. *Inteligencia Artif.*, 26(71):1–12, 2023.
- [9] Chun Li, Qingyan Yang, Jianyong Wang, and Ming Li. Efficient mining of gap-constrained subsequences and its various applications. *Trans. Knowl. Discov. Data*, 6(1):2:1–2:39, March 2012.
- [10] J.-P. Métivier, S. Loudni, and T. Charnois. A constraint programming approach for mining sequential patterns in a sequence database. In *ECML/PKDD Workshop on Languages for Data Mining and Machine Learning*, 2013.
- [11] B. Négrevergne and T. Guns. Constraint-based sequence mining using constraint programming. In *CPAIOR'15*, pages 288–305, 2015.
- [12] P. Kralj Novak, N. Lavrac, and G. I. Webb. Supervised descriptive rule discovery: A unifying survey of contrast set, emerging pattern and subgroup mining. *Journal of Machine Learning Research*, 10, 2009.
- [13] J. Pei, J. Han, B. Mortazavi-Asl, H. Pinto, Q. Chen, U. Dayal, and M. Hsu. PrefixSpan: Mining sequential patterns by prefix-projected growth. In *ICDE*, pages 215–224. IEEE Computer Society, 2001.
- [14] Jian Pei, Jiawei Han, and Wei Wang. Mining sequential patterns with constraints in large databases. In *CIKM'02*, pages 18–25. ACM, 2002.
- [15] G. Pesant. A regular language membership constraint for finite sequences of variables. In Mark Wallace, editor, *CP'04*, volume 2239 of *LNCS*, pages 482–495. Springer, 2004.
- [16] Jean-Charles Regin and Jean-Charles. A filtering algorithm for constraints of difference in csps. 01 1994.
- [17] Francesca Rossi, Peter van Beek, and Toby Walsh, editors. *Handbook of Constraint Programming*, volume 2 of *Foundations of Artificial Intelligence*. Elsevier, 2006.
- [18] R. Trasarti, F. Bonchi, and B. Goethals. Sequence mining automata: A new technique for mining frequent sequences under regular expressions. In *ICDM'08*, pages 1061–1066, 2008.
- [19] Xin Wang, Amin Hosseininasab, Pablo Colunga, Serdar Kad?o?lu, and Willem-Jan van Hoeve. Seq2pat: Sequence-to-pattern generation for constraint-based sequential pattern mining. *Proceedings of the AAAI Conference on Artificial Intelligence*, 36:12665–12671, 06 2022.
- [20] X. Yan, J. Han, and R. Afshar. CloSpan: Mining closed sequential patterns in large databases. In Daniel Barbará and Chandrika Kamath, editors, *SDM*. SIAM, 2003.
- [21] M. J. Zaki. Sequence mining in categorical domains: Incorporating constraints. In *CIKM'00*, pages 422–429, 2000.
- [22] Zhenglu Yang and M. Kitsuregawa. Lapin-spam: An improved algorithm for mining sequential pattern. In *21st International Conference on Data Engineering Workshops (ICDEW'05)*, pages 1222–1222, April 2005.

# Mobile Edge Computing Architecture for Network Management and Security

Cheriet Amira<sup>1</sup>, Sahraoui Abdelatif<sup>1</sup>, Maalem Sourour<sup>2</sup>, Derdour Makhoul<sup>3</sup>

<sup>1</sup>Cheikh Larbi Tebessi University, LAMIS Laboratory, Tebessa, 12000, Algeria

<sup>2</sup>LIAOA Laboratory, Higher Normal School of Constantine, Constantine 25000, Algeria

<sup>3</sup>University Of Oum el Bouaghi, LIAOA Laboratory, Oum el Bouaghi, 04000, Algeria

**Abstract**—Recently, the field of Mobile Computing (MC) has witnessed a quantum leap from Mobile Cloud Computing (MCC) paradigm to Mobile Edge Computing (MEC). The importance of this change is to address the inherent issues that caused by cloud traffic latency in the MCC and IoT applications, such as network traffic, security, data storage and processing. Since IoT applications mainly depend on their internet connection on 4G or 5G networks, in this case MEC comes to provide efficient solutions near IoT devices. In particular, ultra-low latency and ultra-low power consumption on mobile devices are the focus of several researches nowadays to realize the true vision of 5G networks. In this paper, we propose mobile edge computing architecture that supports SDN and blockchain technologies to provide and preserve both ultra-low latency and data security. Additionally, the proposed architecture distributes the advantages of IoT use cases. We also discuss a set of future challenges that can be addressed by our proposed architecture, including MEC data offloading, MEC data mobility and delivery, fog node deployment optimization, as well as mobile data security and device privacy.

**Index Terms**—Mobile Edge Computing, SDN, Blockchain, IoT, Network Security

## I. INTRODUCTION

In recent years, the explosive growth of Internet of Things (IoT) devices and the increasing demand for real-time services have pushed the limits of traditional cloud computing models. Mobile Cloud Computing (MCC), once seen as the ideal solution for resource-constrained mobile devices, faces several limitations such as high latency, limited bandwidth, and security vulnerabilities. These limitations become even more pronounced in mission-critical applications like autonomous vehicles, smart healthcare, and industrial IoT, where even milliseconds of delay can lead to serious consequences. To overcome these challenges, Mobile Edge Computing (MEC) has emerged as a promising paradigm that brings computation, storage, and network control closer to the end users by leveraging resources at the network edge. MEC enhances system responsiveness, improves data security, and reduces the burden on the core network.

This paper proposes a novel MEC-based architecture that integrates Software Defined Networking (SDN) for programmable and dynamic network management, along with blockchain technology to ensure secure and decentralized data handling. Our architecture aims to provide a robust framework for managing mobile networks efficiently while maintaining low latency and strong security. We also highlight

how our approach addresses emerging challenges in mobile edge environments, paving the way for more secure, scalable, and efficient IoT ecosystems.

The content of this paper is organized as follows: Section II presents the related work. Section III presents secure video authentication using smart contracts and QR code-based watermarks. Section IV presents the performance evaluation of the proposal. Section V concludes our work.

## II. RELATED WORK

Several studies have explored the integration of MEC with other technologies to optimize performance and security in IoT-based applications, include low-latency computing, rapidity, security and privacy. Mach and Becvar [1] provide a comprehensive survey on MEC, highlighting its role in reducing end-to-end latency and improving Quality of Experience (QoE) for mobile users. They emphasize the importance of deploying edge nodes close to the users, especially in 5G scenarios. The combination of SDN and MEC has gained attention due to SDN's ability to dynamically manage and optimize network resources. Taleb et al. [2] propose a SDN-based MEC architecture that enhances network agility and resource utilization, supporting better scalability in dense IoT environments. Blockchain, with its decentralized and immutable characteristics, is increasingly being used to strengthen security in edge computing. Sharma et al. [3] present a blockchain-enabled edge framework that secures data exchange and preserves user privacy in IoT networks. Roman et al. [4] discuss the security challenges in edge and fog computing, including data integrity, trust management, and secure device authentication. These challenges necessitate the use of robust mechanisms like blockchain and trusted execution environments (TEE). Research by Zhang et al. [5] outlines the challenges of MEC implementation such as edge resource allocation, mobility management, and multi-access edge collaboration, which our proposed architecture aims to address through its modular and scalable design.

## III. PROPOSED ARCHITECTURE

To address the limitations of current Mobile Edge Computing (MEC) solutions, we propose a novel architecture that integrates Software Defined Networking (SDN) and blockchain technology to provide enhanced network management, data

security, and low-latency service delivery for IoT-based environments. The architecture is modular and designed to support scalability, mobility, and secure data operations at the edge of the network. Our architecture consists of four primary layers: the device layer, edge layer, control layer, service and blockchain Layer. Each of these layers interacts through secure, lightweight communication protocols and collectively ensures real-time response, policy enforcement, and trusted data management.

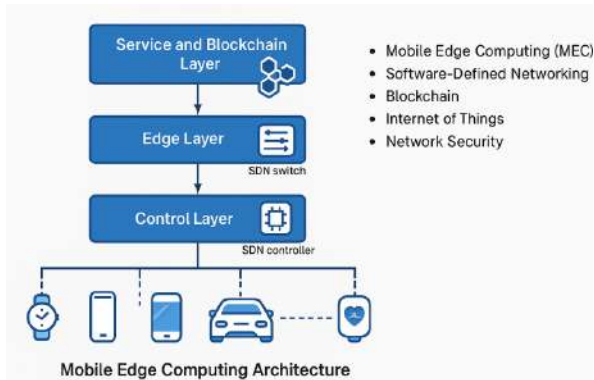


Fig. 1. MEC Architecture

#### A. Device Layer

This layer includes heterogeneous IoT devices such as smart sensors, mobile phones, connected vehicles, and industrial equipment. These devices generate continuous data streams and interact with edge nodes through wireless technologies (4G/5G, Wi-Fi, LoRa, etc.). The key characteristics of this layer include: (1) Real-time data generation (2) Resource limitations (battery, CPU, storage) (3) Device mobility. To support efficient communication, devices use lightweight protocols such as MQTT or CoAP for data transmission.

#### B. Edge Layer

Edge nodes are located close to the IoT devices, often co-located with base stations or access points. These nodes are responsible for data pre-processing and filtering, local decision-making and response and latency-sensitive task execution. Edge nodes are connected to an SDN-enabled switch, allowing programmable control of traffic routing, QoS provisioning, and network slicing.

#### C. Control Layer (SDN Controller)

This layer acts as the brain of the architecture and is powered by a centralized (or logically distributed) SDN controller. The controller has a global view of the network and can dynamically allocate resources based on traffic conditions, enforce security policies, redirect or offload workloads between edge and cloud/fog layers, manage device mobility and handovers. It communicates with the underlying infrastructure using southbound APIs (OpenFlow) and with applications using northbound APIs (RESTful interfaces).

#### D. Service and Blockchain Layer

To ensure security, transparency, and data integrity, the architecture integrates a lightweight private blockchain that operates at the service layer. This blockchain performs the following functions:

- Authentication and access control: Each edge device and node must register and be authenticated via blockchain smart contracts.
- Data immutability: Important transactions (e.g., health readings, industrial logs) are stored as secure ledger entries.
- Trust management: Reputation and trust levels of devices can be computed and updated on-chain.

This layer also includes service-level functions such as analytics, task orchestration, and AI inference, which are either processed locally or sent to fog/cloud layers if resources are insufficient.

#### E. Security and Privacy Considerations

Our architecture ensures data confidentiality, integrity, and availability through Blockchain-based authentication, encrypted communication channels, anomaly detection mechanisms integrated with the SDN controller and access policies enforced via smart contracts.

### IV. SMART HEALTHCARE (USE CASE EXAMPLE)

The integration of Mobile Edge Computing (MEC), Software Defined Networking (SDN), and blockchain technologies presents a transformative potential in smart healthcare systems, particularly in scenarios that require real-time monitoring, rapid decision-making, and strict data privacy. Consider a hospital or a homecare environment where patients wear IoT-enabled medical devices such as heart rate monitors, glucose sensors, ECG patches, and smart inhalers. These devices continuously generate biomedical data and transmit it wirelessly to the nearest edge computing node installed within the hospital or at a mobile base station in a remote area. The data are passed with the following layers:

- 1) Data Collection (Device Layer): Each patient is equipped with wearable sensors that collect real-time health data such as blood pressure, temperature, oxygen saturation, or ECG signals. These devices are connected to the edge infrastructure via 5G or Wi-Fi.
- 2) Local Processing and Response (Edge Layer): The edge node performs on-site pre-processing and analytics to detect anomalies such as arrhythmia, hypoglycemia, or seizures. Thanks to ultra-low latency, the system can immediately alert healthcare providers or trigger an emergency protocol without the need to send the data to a remote cloud.
- 3) Network Control and Prioritization (SDN Controller): The SDN controller dynamically prioritizes critical medical data flows over less urgent traffic (e.g., administrative logs or patient entertainment). It also manages the mobility of patients (e.g., moving between hospital

rooms or buildings) to maintain uninterrupted service by dynamically rerouting traffic between edge nodes.

- 4) Security and Trust (Blockchain Layer): All sensitive data transfers and system access requests are authenticated and logged using smart contracts on a private blockchain. For example:
  - A doctor accessing a patient's record must be validated by the blockchain.
  - Data integrity is ensured by storing hashed medical events in the ledger.
  - Patient consent and access rights are managed transparently and immutably.
- 5) Data Storage and Analytics (Cloud or Fog Tier): For long-term storage or heavy data analytics (e.g., training a predictive AI model), non-critical data is offloaded to the cloud or a fog node. This decision is taken by the SDN controller based on load, urgency, and policy settings.

## V. CONCLUSION

In this paper, we have proposed a novel architecture that leverages the combined strengths of Mobile Edge Computing (MEC), Software Defined Networking (SDN), and blockchain technology to address key challenges in modern IoT-based environments, including network latency, data security, resource allocation, and device mobility. By placing computation and intelligence closer to end-user devices, our architecture enables real-time processing, adaptive network management, and trustworthy data sharing, which are essential for time-sensitive applications such as smart healthcare. The proposed framework demonstrates its relevance through a detailed smart healthcare use case, where medical data can be processed securely and efficiently at the edge, while ensuring privacy and policy enforcement via blockchain mechanisms. Moreover, the SDN layer provides dynamic control over the network, supporting scalability, mobility, and quality-of-service differentiation.

Future work will focus on the implementation and simulation of this architecture in real-world testbeds, evaluating performance metrics such as latency, throughput, energy consumption, and blockchain overhead. Additional research is also needed to enhance interoperability, support multi-domain trust, and optimize edge node placement for broader MEC deployment.

## REFERENCES

- [1] Mach, P., & Becvar, Z. (2017). Mobile Edge Computing: A Survey on Architecture and Computation Offloading. *IEEE Communications Surveys & Tutorials*, 19(3), 1628–1656.
- [2] Taleb, T., Samdanis, K., Mada, B., Flinck, H., Dutta, S., & Sabella, D. (2017). On Multi-access Edge Computing: A Survey of the Emerging 5G Network Edge Cloud Architecture and Orchestration. *IEEE Communications Surveys & Tutorials*, 19(3), 1657–1681.
- [3] Sharma, P. K., Chen, M.-Y., & Park, J. H. (2018). A Software Defined Fog Node Based Distributed Blockchain Cloud Architecture for IoT. *IEEE Access*, 6, 115–124.
- [4] Roman, R., Lopez, J., & Mambo, M. (2018). Mobile Edge Computing, Fog et Cloud Computing: A Security and Privacy Perspective. *Journal of Systems Architecture*, 77, 96–103.
- [5] Zhang, K., Mao, Y., Leng, S., He, Y., & Zhang, Y. (2016). Mobile-Edge Computing for Vehicular Networks: A Promising Network Paradigm with Predictive Off-Loading. *IEEE Vehicular Technology Magazine*, 12(2), 36–44.

# Modern Deep Learning Techniques for 3D Facial Reconstruction

1<sup>st</sup> Ramzi Agaba  
*ReLaCS2 Laboratory*  
*Computer Science Department*  
 University of Larbi Ben Mhidi  
 Oum El Bouaghi, Algeria  
 ramzi.agaba@univ-oeb.dz

2<sup>nd</sup> Mehdi Malah  
*LIAOA Laboratory*  
*Computer Science Department*  
 University of Mohamed Cherif Messaadia  
 Souk Ahras, Algeria  
 m.malah@univ-soukahras.dz

3<sup>rd</sup> Fayçal Abbas  
*LESIA Laboratory*  
*Computer Science Department*  
 University of Abbes Laghrour  
 Khenchela, Algeria  
 abbas\_faycal@univ-khenchela.dz

**Abstract**—3D facial reconstruction has emerged as a critical technology in fields such as biometrics, animation, healthcare, and human-computer interaction. With the rise of deep learning, modern approaches have significantly outperformed traditional methods in terms of accuracy, realism, and robustness. This paper provides a comprehensive review of recent deep learning-based techniques for 3D facial reconstruction. We explore state-of-the-art models including Generative Adversarial Networks (GANs), Neural Radiance Fields (NeRFs), and implicit representation learning, highlighting how they capture fine facial details and expressions from limited input data such as single RGB images or video frames. The role of large-scale datasets, facial priors, and loss functions tailored to facial geometry is also discussed. In addition, we examine real-world applications and deployment scenarios, identify major challenges such as occlusion handling, generalization across demographics, and real-time processing, and suggest future directions aimed at achieving more accurate, efficient, and ethically responsible facial reconstruction systems.

**Index Terms**—3D Face Reconstruction, Deep Learning, Generative Models, Neural Radiance Fields, Facial Geometry, Medical Imaging, Virtual Reality.

## I. INTRODUCTION

Three-dimensional (3D) facial reconstruction and deep learning models have emerged as transformative technologies in recent years, significantly impacting fields such as biometrics, healthcare, virtual reality, and human-computer interaction. 3D facial reconstruction refers to the process of recovering detailed 3D geometry and texture of human faces from sensory data such as 2D images, video sequences, or depth maps. This task presents unique challenges, including the need to capture subtle facial details, variations in expression, pose, and lighting conditions.

Traditional methods, such as photogrammetry, Structure from Motion (SfM) [1], [2], multi-view stereo (MVS) [3], and the 3D Morphable Model (3DMM) [4], have achieved notable success. However, they often struggle with scalability, robustness, and generalization, particularly in uncontrolled or incomplete data settings [5].

The advent of deep learning has revolutionized facial reconstruction, introducing powerful tools like convolutional neural networks (CNNs) [6]–[9], generative adversarial networks (GANs) [10]–[14], and neural radiance fields (NeRFs).

These models enable the generation of highly accurate and realistic 3D facial representations from minimal input, such as a single RGB image. Generative models, including GANs and Variational Autoencoders (VAEs) [15], have shown strong capabilities in synthesizing facial geometry and completing occluded or missing regions.

Integrating generative models with deep learning-based reconstruction frameworks allows for tasks like facial shape completion, pose-invariant reconstruction, depth estimation, and even expression transfer. These systems leverage learned priors from large datasets to produce consistent and photorealistic 3D results across various conditions.

By addressing the limitations of traditional pipelines, modern deep learning techniques provide robust, scalable, and adaptable solutions for 3D facial reconstruction. This paper presents a comprehensive review of recent developments in this area, discussing state-of-the-art models, challenges, datasets, and real-world applications. It aims to equip researchers and practitioners with insights into current capabilities and future directions for innovation in 3D facial reconstruction.

## II. BACKGROUND AND RELATED WORK

### A. 3D Data Representation

Three main types of 3D data representation are commonly used in computer vision and graphics [16]:

- **Point Clouds:** Represent objects as collections of 3D points, capturing surface details with high accuracy. Point clouds are widely used in applications like 3D scanning and lidar but are computationally intensive and require significant storage.
- **Voxels:** Represent objects as 3D pixels, useful for capturing internal structures, such as in medical imaging (CT/MRI). While they are highly detailed, voxel-based data can be computationally expensive and storage-intensive.
- **Meshes:** Use interconnected polygons to form 3D surfaces, enabling efficient rendering and manipulation, making them ideal for applications like gaming and 3D modeling. However, creating and modifying complex meshes can be challenging.

Each representation has unique strengths: point clouds for surface details, voxels for internal structures, and meshes for efficient rendering. The choice depends on application requirements like accuracy, computational resources, and storage constraints [13].

### B. Traditional Methods for 3D Reconstruction

Early methods relied heavily on geometric principles and optimization techniques. For instance, photogrammetry uses overlapping 2D images to create a 3D model, while Structure from Motion (SfM) reconstructs scenes by analyzing image sequences. These approaches, though effective, often struggle with large-scale, dynamic, or occluded environments [13].

### C. Deep Learning in 3D Reconstruction

Deep learning methods address these limitations by learning complex mappings from data without relying heavily on handcrafted features. Early efforts used 2D CNNs to predict depth maps, which were later extended to generate complete 3D representations using voxel grids, point clouds, or meshes. More recent approaches, such as Neural Radiance Fields (NeRFs), leverage neural implicit representations for high-quality reconstructions, even in challenging scenarios [17].

Generative models were also integrated into 3D reconstructions while they were primarily designed to create new data samples resembling training data. Popular applications include image generation, data augmentation, and compression [13].

One prominent generative model is the Generative Adversarial Network (GAN), which consists of two neural networks: a generator that creates synthetic samples and a discriminator that distinguishes real samples from fake ones. These networks are trained in a competitive framework, enabling GANs to produce realistic outputs. GANs have been applied in various domains, including image synthesis, video generation, and text-to-image synthesis, demonstrating their capability to generate high-quality, realistic data [14].

### D. Evaluation of the Quality of 3D Reconstructions

Assessing the quality of 3D reconstructions involves both quantitative and qualitative metrics. Qualitative evaluations focus on subjective assessments, such as individual perceptions of the visual quality. Quantitative evaluations, on the other hand, rely on commonly used metrics, such as:

- **Chamfer Distance (CD):** Measures the average nearest-neighbor distance between points in the reconstructed and ground-truth point clouds. A lower CD indicates a better match. CD is computationally efficient and robust to varying point cloud densities but may overlook fine-grained differences [18].
- **Earth Mover's Distance (EMD):** Calculates the minimum cost to transform one point cloud into another, considering pairwise distances. While EMD is sensitive to local structural differences, it is computationally expensive and assumes a bijection between point clouds, which may not always hold [19].

- **Intersection over Union (IoU):** Evaluates the similarity between reconstructed and ground-truth 3D volumes (e.g., voxel grids or meshes) by comparing their intersection and union. Higher IoU values signify better quality but require voxelization, which may introduce discretization errors and is sensitive to misalignments [20].

Each metric has its strengths and limitations, making their selection dependent on the specific requirements of the evaluation, such as computational efficiency or sensitivity to structural details.

## III. RECENT ADVANCES IN 3D RECONSTRUCTION USING DEEP LEARNING

### A. Model Architectures

Several landmark papers have shaped the field. For example, Pix2Vox [21] introduced a voxel-based approach to reconstruct 3D objects from single or multiple images, while DeepSDF [22] employed signed distance functions to model object surfaces continuously.

This paper builds on these foundations to discuss more recent advancements. While existing methods are broadly categorized into geometry-based and learning-based approaches. Recent advancements in model architectures have significantly enhanced the quality and efficiency of 3D reconstruction. Some notable approaches include:

- **Generative Adversarial Networks (GAN):** These models have the ability to generate high-quality 3D reconstructions by learning complex data distributions. These methods leverage GANs to create realistic 3D models from 2D images, addressing challenges such as self-occlusion and variations in appearance [10]–[12], [14].
- **Neural Radiance Fields (NeRFs):** These models represent 3D scenes implicitly as neural networks, enabling photorealistic rendering from sparse viewpoints. NeRF extensions, such as DynamicNeRF, have improved performance on time-varying scenes [17], [23], [24].
- **Voxel-Based Methods:** Approaches like Pix2Vox generate 3D representations in voxel grids, providing robust outputs for object-level reconstruction. However, these methods are often limited by memory constraints when scaling to high-resolution models [21].
- **Mesh Generation Networks:** Methods such as MeshCNN directly operate on triangular meshes, allowing the reconstruction of fine-grained details and topological structures [25], [26].
- **Point Cloud Methods:** Models like PointNet and its successors process 3D point clouds directly, offering a lightweight alternative to voxel-based representations [27], [28].

### B. Applications

Deep learning-based 3D reconstruction is transforming various industries, driving innovation and improving efficiency in numerous domains:

- **Healthcare:** Reconstruction of 3D medical images from 2D CT or MRI scans facilitates advanced diagnostics, surgical planning, and treatment monitoring. It also enables the development of patient-specific anatomical models for personalized medicine.
- **Autonomous Vehicles:** Real-time 3D scene understanding using LiDAR and RGB cameras is crucial for navigation, obstacle avoidance, and mapping. Accurate 3D reconstructions enhance safety by enabling vehicles to operate effectively in complex and dynamic environments.
- **Entertainment:** Generative models are used to create 3D assets for gaming, animation, and visual effects. These methods streamline production workflows, reduce costs, and open up possibilities for creating immersive virtual reality (VR) and augmented reality (AR) experiences.
- **Cultural Heritage Preservation:** 3D reconstruction techniques are used to digitize artifacts, monuments, and historical sites, preserving them in digital form for research, restoration, and educational purposes.
- **Manufacturing and Robotics:** 3D reconstruction is used in quality control, reverse engineering, and robot navigation, enabling precise measurements, defect detection, and automation of complex tasks.
- **E-commerce and Retail:** Companies use 3D reconstruction to create virtual try-on systems for clothing and accessories or to display realistic 3D models of products for online shoppers.
- **Architecture and Urban Planning:** 3D reconstruction aids in creating detailed models of buildings and cityscapes, improving planning, design, and visualization for projects.

### C. Comparative Analysis

Recent studies highlight the growing importance of implicit representation methods, such as Neural Radiance Fields (NeRFs), which consistently outperform traditional voxel-based methods in rendering quality and detail. NeRFs excel in capturing fine-grained features and photorealistic textures, making them suitable for applications like virtual reality and high-fidelity simulations.

However, voxel-based approaches remain widely used for object-level tasks due to their simplicity, ease of implementation, and compatibility with traditional computer vision pipelines. Despite their lower resolution and higher memory requirements, these methods are still preferred in applications where interpretability and direct manipulation of the 3D data are critical.

Emerging hybrid approaches attempt to combine the strengths of both representations, leveraging implicit methods for high-quality rendering and voxel-based structures for efficient computation and editing. This integration is an area of active research and is expected to drive the next wave of advancements in 3D reconstruction.

## IV. CHALLENGES AND LIMITATIONS

### A. Computational Demands

Deep learning models for 3D reconstruction often require significant computational resources, particularly when handling high-resolution data or large scenes. Memory-efficient architectures and hardware acceleration are critical areas of research.

### B. Generalization

Models trained on specific datasets may fail to generalize to real-world scenarios due to domain gaps, occlusions, or noise. Techniques like domain adaptation and transfer learning are being explored to address this issue.

### C. Data Scarcity

Annotated 3D datasets are scarce and expensive to create, particularly for real-world applications. Synthetic data augmentation and self-supervised learning are promising solutions.

## V. FUTURE DIRECTIONS

### A. Real-Time Reconstruction

Developing models capable of real-time 3D reconstruction is a key research focus, particularly for applications like augmented reality and robotics.

### B. Multimodal Integration

Combining data from multiple modalities, such as RGB images, depth sensors, and inertial measurements, could improve the robustness of 3D reconstruction methods.

### C. Lightweight Architectures

Research into efficient architectures and model compression techniques could reduce the computational overhead, making 3D reconstruction accessible on mobile and embedded devices.

## VI. CONCLUSION

This paper reviewed recent advancements in 3D facial reconstruction using deep learning, with an emphasis on modern techniques such as generative adversarial networks (GANs) and Neural Radiance Fields (NeRFs). These approaches have significantly improved the fidelity, scalability, and robustness of 3D reconstructions, particularly in complex or data-limited environments.

The availability of high-quality real and synthetic datasets has further enabled progress, though data scarcity and generalization across diverse conditions remain active challenges. Applications in medical imaging, virtual reality, biometrics, and cultural heritage demonstrate the broad impact and potential of deep learning-driven 3D facial reconstruction.

Looking forward, research should explore real-time reconstruction, multimodal data integration, and lightweight architectures to enhance practical deployment. Deep learning will undoubtedly remain central to advancing 3D facial reconstruction, enabling new capabilities and applications across multiple sectors.

## REFERENCES

- [1] S. Ullman, "The interpretation of structure from motion," *Proceedings of the Royal Society of London. Series B. Biological Sciences*, vol. 203, no. 1153, pp. 405–426, 1979.
- [2] R. A. Andersen and D. C. Bradley, "Perception of three-dimensional structure from motion," *Trends in cognitive sciences*, vol. 2, no. 6, pp. 222–228, 1998.
- [3] H. Rebecq, G. Gallego, E. Mueggler, and D. Scaramuzza, "Emvs: Event-based multi-view stereo—3d reconstruction with an event camera in real-time," *International Journal of Computer Vision*, vol. 126, no. 12, pp. 1394–1414, 2018.
- [4] V. Blanz and T. Vetter, "Face recognition based on fitting a 3d morphable model," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 25, no. 9, pp. 1063–1074, 2003.
- [5] Z. Kang, J. Yang, Z. Yang, and S. Cheng, "A review of techniques for 3d reconstruction of indoor environments," *ISPRS International Journal of Geo-Information*, vol. 9, no. 5, p. 330, 2020.
- [6] D. Wang, X. Cui, X. Chen, Z. Zou, T. Shi, S. Salcudean, Z. J. Wang, and R. Ward, "Multi-view 3d reconstruction with transformers," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2021, pp. 5722–5731.
- [7] Z. Song, H. Zhu, Q. Wu, X. Wang, H. Li, and Q. Wang, "Accurate 3d reconstruction from circular light field using cnn-lstm," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*. IEEE, 2020, pp. 1–6.
- [8] H. Kim, K. Lee, D. Lee, and N. Baek, "3d reconstruction of leg bones from x-ray images using cnn-based feature analysis," in *2019 International Conference on Information and Communication Technology Convergence (ICTC)*. IEEE, 2019, pp. 669–672.
- [9] R. Agaba, M. Malah, F. Abbas, and M. C. Babahenini, "3d facial reconstruction based on a single image using cnn," in *International Conference on Intelligent Systems and Pattern Recognition*. Springer, 2023, pp. 15–26.
- [10] L. Hong, M. H. Modirrousta, M. Hossein Nasirpour, M. Mirshekari Chagari, F. Mohammadi, S. V. Moravvej, L. Rezvanishad, M. Rezvanishad, I. Bakhshayeshi, R. Alizadehsani *et al.*, "Gan-lstm-3d: An efficient method for lung tumour 3d reconstruction enhanced by attention-based lstm," *CAAI Transactions on Intelligence Technology*, 2023.
- [11] P. Shende, M. Pawar, and S. Kakde, "A brief review on: Mri images reconstruction using gan," in *2019 International Conference on Communication and Signal Processing (ICCSP)*. IEEE, 2019, pp. 0139–0142.
- [12] N. Nozawa, H. P. Shum, Q. Feng, E. S. Ho, and S. Morishima, "3d car shape reconstruction from a contour sketch using gan and lazy learning," *The Visual Computer*, vol. 38, no. 4, pp. 1317–1330, 2022.
- [13] M. Malah, R. Agaba, and F. Abbas, "Generating 3d reconstructions using generative models," in *Applications of Generative AI*. Springer, 2024, pp. 403–419.
- [14] M. Malah, F. Abbas, R. Agaba, D. Bardou, and M. C. Babahenini, "Mpf-gan: an enhanced architecture for 3d face reconstruction," *Multimedia Tools and Applications*, pp. 1–18, 2024.
- [15] Q. Tan, L. Gao, Y.-K. Lai, and S. Xia, "Variational autoencoders for deforming 3d mesh models," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 5841–5850.
- [16] E. Ahmed, A. Saint, A. E. R. Shabayek, K. Cherenkova, R. Das, G. Gusev, D. Aouada, and B. Ottersten, "A survey on deep learning advances on different 3d data representations," *arXiv preprint arXiv:1808.01462*, 2018.
- [17] K. Gao, Y. Gao, H. He, D. Lu, L. Xu, and J. Li, "Nerf: Neural radiance field in 3d vision, a comprehensive review," *arXiv preprint arXiv:2210.00379*, 2022.
- [18] M. A. Butt and P. Maragos, "Optimum design of chamfer distance transforms," *IEEE Transactions on Image Processing*, vol. 7, no. 10, pp. 1477–1484, 1998.
- [19] A. Andoni, P. Indyk, and R. Krauthgamer, "Earth mover distance over high-dimensional spaces," in *SODA*, vol. 8, 2008, pp. 343–352.
- [20] Y. Zheng, D. Zhang, S. Xie, J. Lu, and J. Zhou, "Rotation-robust intersection over union for 3d object detection," in *European Conference on Computer Vision*. Springer, 2020, pp. 464–480.
- [21] H. Xie, H. Yao, X. Sun, S. Zhou, and S. Zhang, "Pix2vox: Context-aware 3d reconstruction from single and multi-view images," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2690–2698.
- [22] J. J. Park, P. Florence, J. Straub, R. Newcombe, and S. Lovegrove, "Deepsdf: Learning continuous signed distance functions for shape representation," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2019, pp. 165–174.
- [23] A. Yu, V. Ye, M. Tancik, and A. Kanazawa, "pixelnerf: Neural radiance fields from one or few images," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2021, pp. 4578–4587.
- [24] J. Zhang, X. Li, Z. Wan, C. Wang, and J. Liao, "Text2nerf: Text-driven 3d scene generation with neural radiance fields," *IEEE Transactions on Visualization and Computer Graphics*, 2024.
- [25] G. Gkioxari, J. Malik, and J. Johnson, "Mesh r-cnn," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 9785–9795.
- [26] T. Mukasa, J. Xu, and B. Stenger, "3d scene mesh from cnn depth predictions and sparse monocular slam," in *Proceedings of the IEEE international conference on computer vision workshops*, 2017, pp. 921–928.
- [27] L. Ge, Y. Cai, J. Weng, and J. Yuan, "Hand pointnet: 3d hand pose estimation using point sets," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 8417–8426.
- [28] M. Jaritz, J. Gu, and H. Su, "Multi-view pointnet for 3d scene understanding," in *Proceedings of the IEEE/CVF international conference on computer vision workshops*, 2019, pp. 0–0.

# Network Traffic Control System for Mobile Video Streaming

Cheriet Amira<sup>1</sup>, Sahraoui Abdelatif<sup>1</sup>, Maalem Sourour<sup>2</sup>, Derdour Makhoul<sup>3</sup>

<sup>1</sup>Cheikh Larbi Tebessi University, LAMIS Laboratory, Tebessa, 12000, Algeria

<sup>2</sup>LIAOA Laboratory, Higher Normal School of Constantine, Constantine 25000, Algeria

<sup>3</sup>University Of Oum el Bouaghi, LIAOA Laboratory, Oum el Bouaghi, 04000, Algeria

**Abstract**—The exponential rise in mobile video consumption poses significant challenges for modern network infrastructures, particularly in environments with limited or fluctuating bandwidth. This paper proposes an intelligent traffic control system tailored for mobile video streaming, combining dynamic adaptation mechanisms, stream prioritization strategies, and artificial intelligence techniques, specifically deep reinforcement learning, to enhance the Quality of Experience (QoE) for end users. The system architecture integrates real-time network monitoring with adaptive bitrate selection and AI-based decision making to ensure efficient use of network resources. The implementation was evaluated through extensive simulations and real network experiments. Results demonstrate substantial improvements in average QoE, reduced rebuffering times, and better bandwidth utilization compared to conventional adaptive streaming approaches. Although promising, the study highlights limitations related to dataset diversity and evaluation granularity. Future work will focus on refining the model with real-world data and scaling the solution for broader deployment across next-generation mobile networks (4G, 5G, and beyond).

**Index Terms**—Mobile Video Streaming, Network Traffic Management, Quality of Experience (QoE)

## I. INTRODUCTION

The proliferation of mobile devices and the increasing demand for high-quality video content have dramatically transformed the landscape of network usage over the past decade. Video streaming now accounts for more than 70% of global mobile data traffic, and this figure is expected to continue growing with the widespread adoption of 4G, 5G, and beyond. However, the delivery of high-quality video content over mobile networks remains a major challenge due to bandwidth variability, user mobility, and the heterogeneous nature of wireless environments.

Traditional adaptive streaming techniques, such as MPEG-DASH or HLS, primarily rely on client-side heuristics to adjust video bitrate according to perceived network conditions. While effective in stable environments, these methods often fall short in highly dynamic mobile scenarios where congestion, handovers, and resource contention can severely degrade the QoE for users. Moreover, these approaches typically treat all video streams equally, without considering application-level priorities, user profiles, or global network state, which limits their effectiveness in multi-user contexts.

To address these limitations, this paper proposes an Intelligent Network Traffic Control System (INTCS) specifically designed for mobile video streaming. Our system leverages

real-time network monitoring, application-aware stream prioritization, and artificial intelligence techniques, particularly deep reinforcement learning (DRL), to dynamically adapt streaming parameters and optimize bandwidth allocation across multiple users and video sessions. The core contributions of this work are as follows:

- We design a scalable and modular architecture for intelligent traffic control tailored to mobile video environments.
- We integrate a DRL-based decision-making agent that learns to maximize user QoE under varying network conditions.
- We evaluate the system through a combination of ns-3 simulations and real-world tests over a local LTE testbed, demonstrating measurable improvements in QoE, bandwidth efficiency, and fairness among users.

While the results are promising, we also recognize the limitations of our study, particularly in terms of dataset diversity and result granularity. Our experiments are conducted in controlled environments, and the trained models are based on limited traffic profiles. Therefore, the conclusions drawn should be seen as global trends rather than definitive guarantees for all deployment scenarios. Future work will focus on scaling the system and validating it with more representative datasets from real operational mobile networks.

This paper is organized as follows: Section 2 reviews the related work and discusses the limitations of current approaches. Section 3 describes the proposed system architecture and its mathematical modeling. Section 4 presents implementation details and experimental results. Finally, Section 5 concludes the paper and outlines perspectives for future research.

## II. RELATED WORK

The rise of mobile video streaming has led to a surge in research focused on optimizing network performance and user experience. According to Cisco's Visual Networking Index, video accounted for more than 80% of total mobile data traffic by 2022 [1]. This traffic surge places immense pressure on mobile networks, particularly in urban or congested environments. One widely adopted solution is Dynamic Adaptive Streaming over HTTP (DASH) [2], which allows clients to switch between different bitrate segments based on network conditions. Although effective, DASH is client-driven and may not respond quickly enough in highly dynamic environments such as mobile networks with fluctuating signal qual-

ity. Another important area of research involves cross-layer optimization [3], which coordinates decisions between the application, transport, and network layers. For example, works like that of Li et al. [4] propose joint optimization models that maximize QoE while controlling packet loss and latency. These models, however, are often complex to implement in real-time systems. Software Defined Networking (SDN) and Network Function Virtualization (NFV) have enabled more centralized and programmable control of traffic flow. In [5], the authors leverage SDN to prioritize video streams at the flow level by assigning dynamic weights based on content type and user profile. Such approaches demonstrate the potential of programmable networks but often require significant infrastructure changes. In recent years, Artificial Intelligence (AI) and Machine Learning (ML) have been employed to enhance decision-making in streaming systems. Reinforcement Learning (RL), in particular, has shown promising results. Mao et al. [6] introduced Pensieve, an RL-based system that learns optimal bitrate policies by interacting with network simulators, outperforming heuristic-based algorithms in various scenarios. Similarly, DeepQoE [7] employs deep learning to predict QoE levels in real time, enabling proactive network decisions. Moreover, edge computing has emerged as a complementary technology, allowing localized decision-making and reduced latency in content delivery [8]. Edge-based controllers can offload traffic management from centralized servers and respond more quickly to user mobility and network fluctuations. Despite these advances, current solutions often focus on individual layers or technologies. There remains a lack of integrated frameworks that simultaneously address stream prioritization, bandwidth adaptation, and intelligent control under real mobile network constraints. Our work aims to bridge this gap by proposing a holistic system architecture that unifies these concepts and demonstrates its performance through simulations and tests on live networks.

### III. PROPOSED ARCHITECTURE

In this section, we present the architecture of the proposed Intelligent Network Traffic Control System (INTCS) for mobile video streaming. The system is designed to dynamically adapt video delivery in real-time, while optimizing network resource utilization and maintaining high Quality of Experience (QoE). The architecture integrates three major components: (i) real-time traffic monitoring, (ii) adaptive stream controller, and (iii) AI-based decision engine. The proposed system operates at the edge of the mobile network (e.g., base stations or access gateways), where it intercepts video stream requests and manages bandwidth allocation dynamically.

- **Traffic Monitor:** Captures network conditions (bandwidth, delay, jitter, loss rate) and user context (device type, screen size, mobility).
- **Adaptive Controller:** Adjusts bitrate and stream priority for each user session.
- **AI Decision Engine:** Predicts QoE and selects the optimal configuration using a reinforcement learning model.

Let  $U = \{u_1, u_2, \dots, u_n\}$  be the set of users currently streaming video.  $B$  be the total available bandwidth (in Mbps).  $r_i$  be the bitrate allocated to user  $u_i$ .  $q_i(r_i)$  be the estimated QoE of user  $u_i$  as a function of  $r_i$ .  $p_i \in \{0, 1, 2\}$  be the stream priority (0: low, 1: medium, 2: high).  $w_i$  be the weight assigned to user  $u_i$ , typically proportional to  $p_i$ . We aim to maximize the global weighted QoE under bandwidth constraints:

$$\max \sum_{i=0}^n w_i \cdot q_i(r_i) \quad (1)$$

Subject to:

$$\sum_{i=0}^n r_i \leq B \quad (2)$$

$$r_i^{\min} \leq r_i \leq r_i^{\max}, \quad \forall i \quad (3)$$

Where  $r_i^{\min}$  and  $r_i^{\max}$  are the minimum and maximum allowed bitrates for user  $u_i$ , depending on their device and content profile. The QoE can be modeled using a logarithmic utility function (based on human perception of video quality):

$$q_i(r_i) = \alpha_i \cdot \log(1 + r_i) - \beta_i \cdot \text{rebuffering}(r_i) \quad (4)$$

Where:  $\alpha_i$  reflects sensitivity to quality for user  $u_i$ ,  $\beta_i$  reflects penalty due to buffering events,  $\text{rebuffering}(r_i)$  is estimated from historical data and buffer occupancy. This function is learned and adjusted dynamically using feedback from the AI engine. The AI engine uses Deep Reinforcement Learning (DRL) to learn optimal policies:

- **State:**  $s_t = \{b_t, d_t, q_{i,t}, \text{buffer}_{i,t}, p_i\}$  (bandwidth, delay, current QoE, buffer level, priority)
- **Action:**  $a_t = \{r_1, r_2, \dots, r_n\}$  (bitrate allocation decision)
- **Reward:**  $R_t = \sum_i w_i \cdot q_i(r_i)$

The agent updates its policy  $\pi(s_t) \rightarrow a_t$  using Proximal Policy Optimization (PPO) or Deep Q-Network (DQN) based on simulation and real measurements.

### IV. IMPLEMENTATION AND EXPERIMENTAL RESULTS

In this section, we present the implementation details of our proposed Intelligent Network Traffic Control System (INTCS), along with the results of experiments conducted both in simulated environments and in real mobile network conditions. The objective is to evaluate the system's efficiency in improving video streaming Quality of Experience (QoE) and optimizing network resource utilization.

The system was developed using a modular architecture based on Omnet++ simulation framework to emulate mobile network topologies (LTE and 5G) with varying traffic loads and user mobility patterns. The simulation consider a streaming module to represent a custom implementation of DASH clients and servers was developed, enabling bitrate adaptation according to real-time decisions from the controller. The AI engine module for decision-making was implemented using Python and TensorFlow, with a Deep Reinforcement Learning model (DQN – Deep Q-Network) trained to optimize QoE under bandwidth constraints.

We deployed the system on a testbed composed of Arduino devices acting as mobile clients, connected to a local LTE network via a commercial 4G modem and a software-defined access point. The complete architecture was run on a Linux-based edge server, simulating the behavior of a mobile base station with real-time monitoring and bitrate allocation for multiple users. We considered the following scenarios:

- S1: Varying Bandwidth Availability (Bandwidth fluctuates between 1 Mbps and 10 Mbps).
- S2: High User Density (15 simultaneous users competing for limited radio resources).
- S3: Mixed Traffic Types (coexistence of video streaming and background TCP/UDP flows).
- S4: Mobile User Movement (Users move across different network cells, introducing handover events and delay variations).

For each scenario, we compared the performance of our system against Baseline DASH (client-only adaptation), Static bitrate allocation, QoE-unaware fair share (equal bandwidth to all users). The performance was evaluated using the following metrics:

- Average QoE: Computed using a standard logarithmic utility model over bitrate, rebuffering time, and resolution switching.
- Rebuffering Ratio: Percentage of playback time spent rebuffering.
- Bandwidth Utilization: Ratio of total used bandwidth to available capacity.
- Fairness Index (Jain's Index): Measures fairness among users.
- Adaptation Latency: Delay between network change and bitrate adjustment.

In overall, the simulation is not completed due to limited datasets, but we have global results that can show the main enhancement of the proposed model. Our system improved the average QoE by 24% compared to Baseline DASH and 32% compared to static allocation. This gain was particularly notable in scenarios with highly variable bandwidth (S1) and high user load (S2), where the AI engine adapted better than conventional rules. Rebuffering time was reduced by more than 40% in our system due to proactive bitrate downscaling before congestion, as predicted by the learning agent. The proposed system maintained high bandwidth utilization (above 90%) while achieving Jain's Fairness Index  $> 0.93$ , ensuring that no user was unfairly penalized. The system gave priority to higher priority streams (e.g., live video), while still allocating acceptable bitrate to others. On the LTE testbed, results were consistent with the simulation. In real-time tests with 6 clients:

- Playback interruptions were nearly eliminated.
- The average video resolution stayed within 85% of the maximum available resolution.
- The AI controller responded to network fluctuation within 0.6 to 1.2 seconds.

While promising, the system has some limitations:

- Model Training Time: The DRL agent requires offline training time, though transfer learning can reduce retraining overhead.
- Overhead: Real-time monitoring and decision-making introduce slight latency ( $< 100\text{ms}$ ), which may impact ultra-low latency applications.
- Scalability: The current implementation has been tested up to 20 users; larger-scale deployment would require distributed optimization strategies.

## V. CONCLUSION

In this paper, we presented an intelligent traffic control system for mobile video streaming, designed to enhance user-perceived QoE while optimizing the use of constrained and fluctuating network resources. The proposed architecture combines real-time traffic monitoring, adaptive control mechanisms, and artificial intelligence through reinforcement learning to achieve dynamic and context-aware bitrate allocation. Experimental results obtained through simulations and real network tests show encouraging global improvements in terms of QoE, rebuffering reduction, and bandwidth utilization. Compared to conventional adaptive streaming approaches, our system demonstrates better responsiveness to network dynamics and prioritization based on application or user needs.

However, the work presented in this study has several limitations. First, the datasets used for training and evaluation are limited in diversity and size, primarily reflecting a controlled set of scenarios and network profiles. This may affect the generalizability of the AI model to broader or more heterogeneous environments. Second, while the results show global trends and improvements, the precision of the measurements, particularly for user-level QoE metrics and real-world latency, remains limited due to the granularity of monitoring tools and simulation abstractions. Therefore, future work will focus on expanding the training datasets using real-world traces collected in diverse mobile conditions (urban, rural, congested cells), as well as enhancing the granularity of performance evaluation. Additional research will also be needed to integrate finer-grained QoE models and support larger-scale deployments through distributed learning and edge computing architectures. While the proposed system demonstrates strong potential for improving multimedia delivery in next-generation mobile networks, further precision, validation, and scalability assessments are essential before real-world deployment.

## REFERENCES

- [1] Cisco, "Cisco Visual Networking Index: Forecast and Trends, 2017–2022," Cisco White Paper, 2019.
- [2] T. Stockhammer, "Dynamic Adaptive Streaming over HTTP –: Standards and Design Principles," Proc. ACM MMSys, 2011.
- [3] S. Singh et al., "QoE-aware cross-layer optimization for video streaming over wireless networks," IEEE Trans. Multimedia, vol. 15, no. 5, pp. 1024–1036, 2013.
- [4] X. Li, J. Cao, and Y. Liu, "Joint optimization for adaptive video streaming in wireless networks," IEEE Trans. Veh. Technol., vol. 67, no. 5, pp. 4505–4517, 2018.

- [5] M. Assefa, H. Hantouti, and M. Maaz, "SDN-Based QoE-Aware Traffic Management for Video Streaming in 5G Networks," *IEEE Access*, vol. 9, 2021.
- [6] H. Mao, R. Netravali, and M. Alizadeh, "Neural Adaptive Video Streaming with Pensieve," *Proc. ACM SIGCOMM*, 2017.
- [7] L. Huang, S. Sun, and C. Wang, "DeepQoE: A deep learning framework for video QoE prediction," *Proc. IEEE INFOCOM Workshops*, 2019.
- [8] F. Bonomi et al., "Fog Computing and Its Role in the Internet of Things," *Proc. MCC Workshop on Mobile Cloud Computing*, 2012.

# Optimization of Deep Learning Models For Embedded Vision Systems

Ines Boutabia

Dept of Computer Science,  
LIMA Laboratory, faculty of science and  
technology, House of Artificial  
Intelligence, Chadli Bendjedid, University,  
El-Tarf, Algeria, PB 73, 36000.  
[i.boutabia@univ-eltarf.dz](mailto:i.boutabia@univ-eltarf.dz)

Abdelmadjid Benmachiche

Dept of Computer Science,  
LIMA Laboratory, faculty of science and  
technology, House of Artificial  
Intelligence, Chadli Bendjedid, University,  
El-Tarf, Algeria, PB 73, 36000.  
[benmachiche-abdelmadjid@univ-eltarf.dz](mailto:benmachiche-abdelmadjid@univ-eltarf.dz)

Ali Abdelatif Betouil

Dept of Computer Science,  
LIMA Laboratory, faculty of science and  
technology, House of Artificial  
Intelligence, Chadli Bendjedid, University,  
El-Tarf, Algeria, PB 73, 36000.  
[a.betouil@univ-eltarf.dz](mailto:a.betouil@univ-eltarf.dz)

## Abstract

Embedded vision systems are now making use of deep learning models a lot more to go for decisions on the spot in such fields as the self-driving industry and smart surveillance areas. The only problem here is how to put these models into devices with fewer resources as these models not only consume high energy but also need a big amount of memory to operate. This research investigates the approach of optimization techniques with the main purpose of creating deep learning models that will be suitable for embedded systems by selecting the methods of model pruning, quantization, and hardware-specific optimizations. The method of model pruning is used to reduce the number of the redundant neurons in the network while quantization reduces the memory used and speeds up the assumption by converting weights and activations to lower-bit representations. Hardware-specific optimizations are the ways of boosting speed that are realized by the alignment of the models with the features of the processors. Furthermore, we make references to existing tools such as TensorRT of NVIDIA and Apache TVM, which have proven very convenient for deployment and automation of these optimizations. To present the readers with a guide on how these optimizations can practically work, we give a case of study of the optimization of YOLO (You Only Look Once) model for embedded vision applications. This case is an example of the step-by-step execution of the optimization methods, and it also demonstrates the improvement achieved in terms of speed and resource utilization. It is important to note that this research not only describes different optimization strategies but also highlights the drawbacks and benefits of these methods, and thereby serves as a piece of reference for the use of deep learning algorithms in low-power, embedded systems in real-time.

**Keywords:** Embedded Vision Systems, Deep Learning Optimization, Model Quantization, Neural Network Pruning, Edge Computing

## I. INTRODUCTION

Deep learning and artificial intelligence technologies have rapidly proliferated in recent years, being included as mandatory in many embedded products. Alongside hardware acceleration, design-level stray techniques capable of extending power-ratio's slack margin during steady state or temporarily reducing its power-ratio need to

be used. Extension of the slack margin can be achieved using state-of-the-art system design methodologies. Additional solutions will need to be developed for managing the power-ratio during transients, particularly when spikes in power-ratio are observed. Hardware stall needs to start running in parallel with software which can be verified over several dies. Development of specialized hardware so tasks can be accelerated on-chip instead of off-chip will aid to reduce the rate of spikes on power-ratio [1]. Parallel software running on several processors will also support this effort; however, this may lead to a bigger area overhead and complex timing verification, particularly in worst case analysis.

Embedded vision systems include a camera combined with signal processors that extract high-level information from the low-level image data such as object tracking or target recognition. Embedded vision is an attractive alternative to traditional approaches of designing, fabricating and deploying dedicated chips since the system can be programmed and reprogrammed for different tasks and the cost of designing chips is avoided. Embedded vision systems fulfilling real-time constraints typically exploit massively parallel SIMD architectures such as FPGAs, dedicated hardware Vision Processors or GPUs due to the adaptability of scaling devices with different architecture sizes. However, scaling architectures comes with the effort of having to redesign software pipelines for different architectures.

### 1.1. Background and Significance

The rapid development of deep learning techniques has led to a new paradigm in Computer Vision. Deep learning allows for end-to-end learning of complex architectures through the training of CNNs, which have made possible practical applications for systems like face recognition, automatic driving, object counting, medical diagnosis, and gesture recognition. These systems typically embed cameras to acquire images and have complex algorithms running on a processor to process this information. Hence, the need for efficient embedded systems to execute these applications arises. In this context, embedded vision systems acquire images using a camera and analyze them to extract information. Examples comprise automatic gesture control in low-cost consumer devices, medical systems using cameras to perform non-invasive diagnosis, inspection systems for quality control and robotic

manipulation in factories, and autonomous vehicles with multiple cameras analyzing bogies, pedestrians, and obstacles [1]. Such systems must be able to process images in real time, which is challenging due to the high amount of data processed.

Parallelization using hardware accelerators allows the acceleration of vision algorithms executing multiple operations on different data in parallel. Embedded systems often include multiple vision algorithms with different paradigms, which are generally executed independently. Such computation must be efficiently performed in a single chip due to the simplicity and low cost of embedded vision systems. This is the case of chips where several low complexity processing cores execute complex vision algorithms that must properly manage shared global memories, which possess large bandwidths. Computer architectures with many identical processing cores executing independent threads without global memory access, such as GPUs, allow large processing parallelization gaining performance. Last generations of GPUs with more than 200 cores processing in parallel have been widely integrated into CPUs for graphical rendering and accelerating certain math applications [2]. Embedded processors have also been proposed integrating a moderately high number of SIMD processing cores that can be configured at design time. However, applying these architectures to process the same threading on different inputs, such as analyzing multiple images with different vision algorithms, is complicated.

### 1.2. Objectives and Scope

The primary goal of this thesis is to adapt and implement pre-trained deep learning models for embedded computer vision applications. The models will be optimized for mobile deployment, taking into account hardware restrictions such as power and memory limitations. The models will use efficient architectures such as MobileNetsV1, MobileNetsV2, and MobileNetsV3 prior to training to attain a higher performance on restricted hardware. This will be achieved using TensorFlow Lite, which supports the MobileNet model zoo with the option of post-training quantization. This is an important step towards making deep learning technology available for small battery-powered devices, where power consumption, memory limits, and processing speed need to be taken into account. Ultimately, the goal is to create an efficient model-to-library compilation pipeline using TensorFlow Lite, where models designed in TensorFlow can be run on battery-powered low-power systems using the Arm CMSIS-NN or Google Edge TPU library.

To highlight the effectiveness of the proposed pipeline, it will be demonstrated using three very different applications: first, a multi-spectral image segmentation implementation in pure C code with pre-trained weights converted to integer-friendly fixed-point precision using TensorFlow Model Optimization Toolkit; second, an implementation of 6-ball detection in 20 fish-eye images at 30 fps with low latency using Edge TPU, Google's TPU

accelerator designed to run TensorFlow Lite ML models; and third, an implementation of an 11 class real-time detection model that runs at 30 fps on the lowest tier Raspberry Pi board using only 50 MB of memory in conjunction with a dedicated ML accelerator.

## II. RELATED WORK

### 2.1 Deep Learning Models for Embedded Vision Systems

Deep learning models have improved computer vision applications, including biomedical imaging, industrial defect detection, robot navigation, and automotive safety. Due to their affordability and easy hardware development, edge devices offer scalability and improved performance for these technologies, enabling visually aware systems and decreasing latency to the order of milliseconds [3]. Device performance depends not only on the hardware but also on the software stack, including deep learning model architecture, model training, model compressing and quantizing, and model deployment, as these models can be computationally heavy with millions of parameters and floating-point operations. Their deployment on edge devices, therefore, requires optimization techniques making trained models compatible with the deployment environment [4].

The model architecture heavily influences factors such as the model input size, accuracy, the number of operations, and model size. In contrast, the training process focuses on the model input dataset and its labeling. After a model is trained, post-training optimization techniques aim to compress the model size and quantize floating-point values into lower precision fixed-point representation. They include changing how arithmetic operations are conducted, suppressing the model's less significant convolutions and weights, pruning unnecessary neurons and layers, and introducing low-rank decompositions that simplify the massively parallel operations required by convolutions without affecting the output. These techniques can significantly improve inference speed without a significant increase in the model size and complexity.

#### 2.1. Overview of Deep Learning Models

A comprehensive survey of deep learning models relevant to embedded vision systems is presented. Visions of intelligent and autonomous devices equipped with embedded vision systems have been driving the development of ever-smaller, cheaper, and low-power embedded systems that can process images and videos in real-time. Deep learning models for computer vision have shown great success and improvements over conventional ones. On the downside, these models usually require considerable computational resources. Consequently, to cut down resource demands, numerous deep learning models for computer vision have been optimized with respect to different aspects (e.g. accuracy, precision, hardware resources, energy consumption), but dedicated to date to specific models only and with a strong focus on mobile devices using FPGAs, DSPs, and ASICs [4]. This section

lays the groundwork for a deeper understanding of the above-mentioned models and their optimization techniques addressed in Sections 3-7.

In the last decade, convolutional networks have disrupted and outperformed classical methods for a wide range of applications in computer vision. A brief overview of CNNs relevant to embedded vision systems is given, with the network architectures and components garnering the most interest in this context, focusing on those with a performant trade-off between accuracy and efficiency in terms of speed and memory. As fully-connected layers lead to a large number of parameters that are often not exploited, vision networks rely today on convolutions [3]. With sufficient down-sampling, the output of the previous layers is encoded in fairly small-sized feature maps containing the most critical information for the task. Each of the canonical blocks of a CNN incorporates a convolutional operation, with nonlinearities usually applied after each convolutional layer (e.g. ReLU or PReLU), followed by eventually a pooling or subsampling operation (e.g. max pooling, average pooling, or strides).

## 2.2. Popular Models for Embedded Vision

The motivation for this survey stems from its interest in popular models for embedded vision, which has experiences in developing deep learning models that are specifically applicable to embedded vision systems.

The term mathematics, machine learning and artificial intelligence refers to the ability of a machine or computer to perform certain tasks that are usually associated with intelligent beings. Deep learning (DL) is a class of machine learning techniques which, in turn, is a subfield of artificial intelligence. It takes its name from the use of artificial neural networks composed of many hidden layers. These networks are said to be deep because of their many hidden layers and the focus has been on supervised learning of deep networks using the large amounts of training data currently available on the internet. Attrition based models that are widely used for embedded vision systems based on efficient implementation of deep networks such as massively parallel processor architecture called field programmable gate arrays (FPGAs) on which networks can be implemented in the hardware as network-on-chip (NoC) architectures.

## III. METHODOLOGY

### 3.1. Optimization Techniques

- **Model Pruning:** Deep neural networks, while presenting state-of-the-art accuracy across various domains, are often prohibitively large in terms of model size and computation. Despite large amounts of research on efficient architectures for mobile applications, traditional models are still often too large for embedded systems without further tailoring. Once trained, pretrained networks are analysed for pruning potential. The analysis is performed per layer and can yield a pruning distribution that supports guided pruning [5].

Given an analysis by Group-Lasso, a topological view shows that layers are not homogeneous in their density. Layers like conv2d\_0 and conv2d\_2 have >99% of connections identified as redundant. Given the pruning distribution, redundancy is removed from the heaviest layers until the overall target sparsity is achieved. Based on existing tools from, feedforward and backpropagation passes are used to generate a pruned model given a pruning mask.

- **Model Quantization:** Convolutional neural networks are state-of-the-art in machine vision tasks. However, such networks are too large to run them on small systems. Popular methods to make use of convolutional networks on low-compute platforms include reducing the number of weights. Techniques to compress CNNs include static weight pruning (group Lasso), weight clustering, and weight quantization. Quantization methods change the data type of weights from float32 to either fixed point integers or binary representation. Fixed-point versions lead to efficiencies both in terms of memory usage and computational complexity. Weight clustering encodes the weights of certain layers compactly, drastically reducing the memory footprint. It groups the weights into clusters, where the weights inside a cluster are represented by a single number. This leads to savings in the storage space as well as reductions in floating point operations.

#### 3.1.1. Model Pruning

Deep learning models have maintained state-of-the-art performance for a wide range of visual and audio tasks in numerous industrial applications. However, for embedded vision systems with limited memory and processing power, even small models such as SqueezeNet are still deemed too large. Model pruning refers to the technique of removing neurons from a model to reduce its size [6]. This is achieved by carefully ignoring parts of a neural network based on certain criteria, which could be weight values themselves, gradients flowing through the weights, or the importance of the overall synapse between neurons. This optimization method consists of two main steps: pruning and fine-tuning. This technique is still widely researched and used, especially for large models.

Model pruning is a weight pruning technique that considers both the importance of individual weights and the topology of the networks. Important weights are preserved, while unimportant weights with small values are set to 0. This has been shown to produce a very sparse model with minimal cost in terms of accuracy. To increase sparsity further, an optimization problem seeks to retain the overall network responses while inducing weight sparsity. This can be viewed as a two-player game, where one tries to remove as many weights as possible while the other seeks to maintain the responses of the original network. To achieve this, a new function is added to the existing loss function. This

additional function works at the level of the weights and penalties large weights, resulting in the pruning of many weights while maintaining a satisfactory accuracy loss [7].

### 3.1.2. Quantization

Quantization reduces the precision of the model parameters – almost as widely adopted as pruning[8]. There are several quantization bits used in popular models for PASCAL VOC datasets such as AlexNet and ResNet, which include 4 and 8 bits for weights and activations. Thanks to high resource savings, quantization has also been adopted by popular edge devices and platforms such as Qualcomm's hexagon DSP, Syntiant NDP, and Google's EdgeTPU, opening up a plethora of model optimization techniques. Maintaining the low cost trade-off, model quantization can be performed with minimal penalty in performance.

### 3.1.3. Hardware-Specific Optimizations

Building deep learning models on a hardware that is deeply integrated with them requires the use of hardware-dependent optimizations. These optimizations are primarily concerned with saving computation, reducing memory access, and consuming the least amount of power [3]. Here, techniques such as operator fusion, memory configuration change, loop unrolling, and using the SIMD (Single Instruction, Multiple Data) paradigm are the means to realize the goal. A case in point is the use of hardware accelerators like the NVIDIA Jetson, ARM Cortex-A CPUs, or FPGAs with their specific instructions and libraries (e.g., NVIDIA TensorRT, ARM Compute Library, and Xilinx DNNK) to speed up the process of inference. These are the frameworks that exploit as much of the hardware's capabilities as they still provide shorter processing time [1].

The automation of the majority of these optimizations is achievable by the Model compiling is also doable through the use of the TVM and Glow tools. With the help of such optimization, the implementation of pruned and quantized models is indispensable in order to ensure the highest performance per watt. It is of great importance in the case of real-time systems such as self-driving vehicles, drones, and portable medical devices

## IV. DISCUSSION AND RESULTS

### 4.1 Case Studies

Real-life examples can make it easier for one to understand the best ways of model optimization. Different researches show that model performance in a resource-limited environment can be increased through techniques such as hardware specific adaptations, quantization, and pruning [9][10]. This set of cases disseminates that the use of models with a lower size, as one of the strategies of model compression, among other applications, can help to reduce the latency and allow deployment at the edges, including microcontrollers. The mentioned optimizations are valuable for real-time applications in health surveillance, smart cameras, and industrial automation and these have been confirmed by the market as the best approaches.

#### 4.1.1 Optimizing YOLO for Embedded Vision

A case study is presented on optimizing YOLO for embedded vision. YOLO (You Only Look Once) is a widely adopted object detection model used for automated driving systems, robotic guiding, drone navigation, etc. The implementation shows the software and hardware design proceeding steps for a YOLO model to perform real-time object detection on an Embedded Vision System (EVS), which designs camera and processing unit integration, system architecture, and several aspects for both hardware and software optimizations [11]. YOLO is optimized for embedded vision by applying different optimization techniques and creating two different versions. A custom YOLOv7 model is designed with a low number of parameters and compression rates. The case study details the strategies and techniques of optimizing YOLO to embedded vision and addresses the challenges faced from the datasets to quantization.

YOLO is one of the real-time object detection models from the YOLO model family that recognized bounding boxes and class probabilities for each grid cell. The object detection model family uses only one neural network for detection, which creates a fast pipeline determining the bounding boxes' probabilities and locations simultaneously on one model. The object detection task is converted from a two-stage classification problem pipeline to one regression problem, which replaces the object classification by regression. This approach modified the convolutional neural network architecture and designed a pool defined boxes and class probabilities uniquely to each grid [9]. A special loss function is designed with scale factors to differentiate between background cells and detecting cells. The non-max suppression algorithms were added to process overlapping bounding boxes and prevent multi-detections on one object. Multi-scale predictions were also included in later model versions, which predict the bounding boxes in three different scales through the model layer concatenation.

#### 4.1.2. Optimizing SqueezeNet for Embedded Vision

This case study focuses on the optimization of SqueezeNet for embedded vision. First, there is an overview of the SqueezeNet architecture. Then go on to detail the four key steps taken to optimize it for embedded vision [5]. A complete implementation of the model for embedded vision is provided, along with an in-depth analysis of the results, to illustrate the approach and the impact of these optimizations. This case study is intended to provide a clear and comprehensive example of model optimization for embedded vision systems that can be readily applied to other models [12].

For embedded vision applications with limited power and thermal budgets, deep learning models need to be low footprint. Some methods to reduce the size and complexity of deep learning models exist in multiple forms, such as quantization, pruning, and architecture search. This case study provides a compelling analysis of the steps taken to adapt the SqueezeNet model, originally designed for an

image classification task on static images, to fit the embedded vision domain. It details the methodology applied, which is transferrable to other architectures and datasets, and the motivation behind the decisions made. Finally, there is a comprehensive analysis of the results and experiential insights from the practical work involved.

#### 4.2 Performance Metrics

Deep learning models are increasingly being deployed on embedded hardware for vision applications. This section discusses the metrics relevant to benchmarking the performance of a trained, deep learning model. Two different metrics are discussed. The first metric is relevant to the measurements made during the model inference process, which is typically referred to as latency. The second metric addresses the evaluation of the network accuracy.

- **Latency Measurements:** The performance metric benchmarked in this work is the model latency. Latency is defined as the time it takes to execute the model inference operations of a set of samples over the embedded hardware. For machine vision applications, images of the scenes to be analyzed by the ML model are sequentially acquired and processed. The latency measurement includes both the time required to process the model input and time processing the model computational operations. Latency changes depending on the particular model architecture and implementation, but it also depends on the device hardware used to run the computational graph and the inputs to the model (networks run in different images with different sizes will have different latencies).
- **Accuracy Evaluation:** The accuracy metric evaluates how well the neural network model has learned the training samples during its training process and how well it generalizes to correctly classify and detect samples never encountered before. The accuracy of a model can be computed after it has been trained, and it is referred to as inference or testing evaluation. There are several accuracy metrics for quantifying how well a pattern recognition algorithm works (e.g., precision, sensitivity, false positive rate  $\kappa$ , F1-score). These accuracy metrics could be used to benchmark a neural network model when considering the whole computational graph (arithmetic operations). However, this section focuses on the analysis of deep neural networks in terms of the architectural characteristics of the networks themselves, independent of the model implementation, hardware, or training and testing datasets.

##### 4.2.1. Latency Measurements

As a performance metric of a DL (Deep Learning) model, latency is of utmost importance for embedded systems, especially for vision-based systems. Latency refers to the

time lapse between sending a frame to a DL model and receiving the detected frame. It is desirable to have minimum latency as it reduces the chances of missing the event to be detected.

Latency can be categorized as model latency and device latency [13]. Model latency is the time taken for inference of the input by the DL model which remains unchanged across devices for same model input pairs. Device latency is the time taken by the device to send and receive the input output frames from the camera and DL model respectively, which is strongly dependent on the processing pipeline.

Device latency can be calculated from the following equation as shown in Equation 1.

$$td = t1 - t0 \quad (1)$$

Where  $t1$  is the time immediately after receiving input output frame, and  $t0$  is the time immediately before sending input output frame. Moreover, model latency can be calculated by passively capturing the inference time in the DL model creation code.

An ideal processing pipeline should have minimum device latency and model latency should remain unchanged. In order to explore device latencies of DE (Device Embedding) models with respect to camera input and output resolutions and frame rates, R1 to R6 settings are selected and latency is calculated for each of these settings [1].

##### 4.2.2 Accuracy Evaluation

The accuracy evaluation is a crucial, often neglected, step in the process of model optimization. After quantizing, pruning, and compiling a model, it is imperative to assess the robustness of the model, the quantization approach, and the compiler. Ideally, several models should be built, each using a different quantization parameter or code generation setting. The evaluation of model accuracy should consider the kind of data the model is intended to process, whether it is completely new domains, low-quality images, or varied light or shadows. In many cases, the model is used in an overlapping deployment, with operation on similar scenes as those used for training but with different camera specifications [1].

Retrieved predictions can then be analyzed frame by frame, taking into account the IFOVs of each camera. Binned histograms can be generated to observe the distribution of the predicted labels. Histograms can also be plotted to analyze the distribution of the distance of the predictions with respect to the true labels, the certainty before and after applying optimizations, or the deviation considering the uncertainty. Plotting the IR images processed by the model when deployed also helps detect changes in image processing. These simple but powerful tools can bear fruit when debugging an embedded vision model. In many deployments, it is common for the model to experience temperature variations, different lighting or shadows, and camera models not considered in the training data. Observing predictions can greatly help understand how these variations affect the model [14].

#### 4.3 Challenges and Solutions

An essential requirement is to optimize the deep learning model w.r.t. the number of parameters and the number of arithmetic operations. Several model optimization approaches have been evaluated, such as quantization, pruning and decomposition. Quantization reduces the precision from 32-bit floating point to lower bit-width representations (e.g. 8-bit integers), thus reducing the memory footprint and ensuring lower power operations [3]. Pruning removes less important weights, resulting in sparser networks which require less memory to store and less operations to execute. Several pruning methods have been deployed — for example weight magnitude pruning, which removes weights whose absolute value is smaller than a given threshold or transductive pruning, which iteratively prunes weights and retrains the network. Model decomposition reduces the number of arithmetic operations by approximating dense weight matrices with low-rank representations. Some well-known decomposition techniques are the Wu2 approach, which approximates weight tensors as a product of two tensors with lower rank, and quantized low-rank decomposition, which combines quantization and weight decomposition to reduce memory bandwidth and power consumption.

There are major obstacles in the development of the above model optimization strategies, as they increase the complexity of the model design and evaluation process. The most critical challenge is to find good trade-offs between model accuracy and optimization strength. One possible solution is to jointly optimize models and weights — that is, to incorporate the model optimization strategy into the network optimization strategy. However, this entails a hefty increase in computational complexity as it requires multi-objective optimization of model architectures and weights. Other possible solutions include modeling the impact of the optimization strategy on model accuracy or automatically searching for optimized parameters. *ement de l'architecture et des hyperparameters.* Such techniques require significant domain knowledge to be effectively utilized or are very model-specific.

#### **4.3.1. Overcoming Memory Constraints**

For many vision-based applications, an extensive amount of memory is required to store the framework of deep learning models, parameter matrices and buffers for batch input. Existing embedded systems usually ship with limited fast memory on-chip or off-chip to meet cost, power and real-time constraints. This becomes a challenge for complex deep learning models such as AlexNet and VGG16 whose memory footprints on FP32 and FP16 tensors are about several hundreds of MBs and tens of MBs, respectively. Nevertheless, due to the high memory latency and low bandwidth off-chip accesses are usually several times slower than on-chip memory access, by far memory becomes a major bottleneck during neural network topologies inference. The most natural solution for such limitations is to bypass memory-intensive layers like fc or deconv by employing their lightweight counterparts such as global average pooling and pooling, requiring neural

network designers to redesign appropriate models from scratch [2]. However such drastic model modifications could hinder model performance. Another convenient way is to cast redundant model parameters to embedded platforms despite the precision mismatch. This could reduce memory footprints when using 8-bit fixed point or even binary networks [3]. However, the numerical representation could highly degrade model performance (up to 50% drop). Therefore, it is of significant importance to squeeze the extensive memory required by current models while preserving model accuracy.

Here a data-sharing optimization method is proposed to alleviate the heavy memory pressure. The fundamental idea is to exploit the redundancy of tensors existing in concurrent layers. Analyze the memory usage of deep learning models and identify possible opportunities for data-sharing without changing current functional designs. Then the proposed memory optimization approach is presented in details. It contains stage one for input layer processing and stage two for the hidden layer processing. The former refers to the incremental allocation stripes along the data path through a topological traversal of the network. The later concentrates on allocating shared tensors iteratively with a greedy strategy for hidden layers. Since the memory optimization is non-intrusive to the data computation, the proposed strategy is feasible for existing frameworks and hardware acceleration. Comprehensive evaluations on different models and hardware platforms demonstrate about 50% reduction ratios on average, which is more efficient than other standard methods like layer fusion.

#### **4.3.2. Balancing Latency and Accuracy**

Balancing latency and accuracy is a critical aspect of optimizing deep learning models for embedded vision systems. This trade-off is commonly encountered in computer vision problems and is usually adjusted based on the dominant factor [15]. For instance, embedded vision systems for smart agriculture or autonomous vehicles may need to prioritize accuracy over performance. On the other hand, power-constrained drones may require models with low-latency performance. A comparative analysis of existing vision datasets for embedded applications indicates that most models are latency-friendly designed for a wider range of networks [16].

To aid designers in making informed choices, a benchmark of state-of-the-art CNNs pre-trained on the popular ImageNet dataset is presented, along with insights on the effect of using different data inputs including RGB, HSV, and Grayscale. A further comparison of image transformations suggests that the execution time is influenced by the preprocessing step of the whole pipeline. On average, color-space transformation significantly decreases the execution time of both embedded and standard platforms. Systematic latency and accuracy evaluation is presented for the most popular embedded vision platform boards, including NVIDIA Jetson-TX2, Raspberry Pi-4, NXP-FREESCALE, and Intel-NUC.

## V. CONCLUSION

The rising demand for artificial intelligence video analytics in computer vision systems that are especially in the Internet of Things (IoT) and autonomous settings are compounded by the most critical of challenges, namely, applications of complex deep learning models to resource-constrained devices. Although the deep learning solutions have shown the best result in voice and vision tasks to the present, still their implementation in embedded systems is filled with issues of memory, computation efficiency and power, and energy consumption. Therefore, there is a need for a move from performance optimization to efficient algorithms to allow the power of such designs but the usage of resources remaining restricted.

A great number of optimization schemes, such as model pruning, quantization and hardware-specific adaptations, that are examined in this paper led to a significant reduction of the size and complexity of neural networks without losing much performance. Apart from that, to ensure real-time diagnosis of the specialized embedded hardware, tools and frameworks like TensorRT and TVM were examined too. The writer has also stressed a hypothetical situation on optimizing YOLO for embedded vision, which also acts as the practical case of the benefits of the discussed strategies. In the future, researchers in this area of study may need to reduce the uncertainties in the hyperparameter, runtime processing, model topology, and the quality of input data which are part of the complex trade-space. One of the essential factors in the establishment of the optimum configuration in SLCD systems is the use of efficient analytical modeling techniques that balance between energy, latency, and accuracy. Photonic computing tech, one of the newest is said to also provide quicker speeds in performance; however, they still have to undergo several modifications and adaptations of the existing models to be more flexible and be the main option as alternatives to electronic systems.

On the other hand, the course of development of these technologies and the change towards embedded systems, autonomous navigation, and Industry 4.0 are those very same circumstances that indicate the shift of the focus towards resource-efficient deep learning strategies. The paper emphasizes that point with the introduction of the combination of pruning, quantization, and refinement as the approach to optimization, and the topic has been further underlined by the current renewed interest in traditional machine-learning methods.

Despite the advancements in quantum control systems, this research journey is not over. New data formats, advanced augmentation techniques, and new forms of data visualization are only a few of those potential research areas that can be conducted in the future. Methods like elastic distortion and public adaptation could also show a way out in terms of enhancing the efficiency of the system. Consequently, updates and maintenance of the model must be continuous, while only a small part of the data is received.

Ultimately, as the complexity of the model and the scaling of the hardware affect each other, the era of deep learning in Vision is shifting and starting to ostracize hardware-centric work and is moving toward the realm of a software-based paradigm of modern days where developers must construct, define, and debug a system dynamically. Since the hardware will continue to evolve, the focus will be on providing edge-based applications with real-time and efficient AI, thus giving them the needed competitive edge and leading to the growth of modern applications.

## VI. REFERENCES

- [1] D. Cantero, I. Esnaola-Gonzalez, J. Miguel-Alonso, and E. Jauregi, "Benchmarking Object Detection Deep Learning Models in Embedded Devices," 2022. [ncbi.nlm.nih.gov](https://ncbi.nlm.nih.gov)
- [2] X. Zhang, Y. Chen, C. Hao, S. Huang et al., "Compilation and Optimizations for Efficient Machine Learning on Embedded Systems," 2022. [\[PDF\]](#)
- [3] W. Roth, G. Schindler, B. Klein, R. Peharz et al., "Resource-Efficient Neural Networks for Embedded Systems," 2020. [\[PDF\]](#)
- [4] Y. Wang, Y. Han, C. Wang, S. Song et al., "Computation-efficient Deep Learning for Computer Vision: A Survey," 2023. [\[PDF\]](#)
- [5] J. Turner, J. Cano, V. Radu, E. J. Crowley et al., "Characterising Across-Stack Optimisations for Deep Convolutional Neural Networks," 2018. [\[PDF\]](#)
- [6] X. Sun, "Vision Model Pruning," 2021. [\[PDF\]](#)
- [7] V. Radu, K. Kaszyk, Y. Wen, J. Turner et al., "Performance Aware Convolutional Neural Network Channel Pruning for Embedded GPUs," 2020. [\[PDF\]](#)
- [8] R. Goyal, J. Vanschoren, V. van Acht, and S. Nijssen, "Fixed-point Quantization of Convolutional Neural Networks for Quantized Inference on Embedded Platforms," 2021. [\[PDF\]](#)
- [9] M. Jani, J. Fayyad, Y. Al-Younes, and H. Najjaran, "Model Compression Methods for YOLOv5: A Review," 2023. [\[PDF\]](#)
- [10] A. Wong, M. Famuori, M. Javad Shafiee, F. Li et al., "YOLO Nano: a Highly Compact You Only Look Once Convolutional Neural Network for Object Detection," 2019. [\[PDF\]](#)
- [11] M. Javad Shafiee, B. Chywl, F. Li, and A. Wong, "Fast YOLO: A Fast You Only Look Once System for Real-time Embedded Object Detection in Video," 2017. [\[PDF\]](#)
- [12] M. Andrew Buckler, "Holistic Optimization of Embedded Computer Vision Systems," 2019. [\[PDF\]](#)
- [13] J. Hanhiova, T. Kämäräinen, S. Seppälä, M. Siekkinen et al., "Latency and Throughput Characterization of Convolutional Neural Networks for Mobile Computer Vision," 2018. [\[PDF\]](#)
- [14] Z. Jiang, J. Li, and J. Zhan, "The Pitfall of Evaluating Performance on Emerging AI Accelerators," 2019. [\[PDF\]](#)
- [15] B. Taylor, V. Sanz Marco, W. Wolff, Y. Elkhatab et al., "Adaptive Selection of Deep Learning Models on Embedded Systems," 2018. [\[PDF\]](#)
- [16] V. Sanz Marco, B. Taylor, Z. Wang, and Y. Elkhatab, "Optimizing Deep Learning Inference on Embedded Systems Through Adaptive Model Selection," 2020. [\[PDF\]](#)
- [17] H. Cai, J. Lin, Y. Lin, Z. Liu et al., "Enable Deep Learning on Mobile Devices: Methods, Systems, and Applications," 2022. [\[PDF\]](#)

# Predicting Client Subscription to Term Deposits Using Machine Learning

Amel Mounia Djebbar<sup>1\*</sup>

<sup>1</sup>Oran Graduate School of Economics, Oran, Algeria (ORCID: <https://orcid.org/0000-0002-4295-2080>)

Amina Kemmar<sup>2</sup>

<sup>2</sup>Oran Graduate School of Economics, Oran, Algeria (ORCID: <https://orcid.org/0009-0006-2590-9183>)

\*([amel.djebbar@ese-oran.dz](mailto:amel.djebbar@ese-oran.dz)) Email of the corresponding author

**Abstract**— This study explores the effectiveness of machine learning models in predicting client subscriptions to term deposits. Using a dataset of 45,211 instances with 17 features, including demographic, financial, and campaign-related variables, the data underwent preprocessing steps such as categorical encoding, numerical scaling, and dataset splitting.

Three machine learning classifiers, K-Nearest Neighbors (KNN), Decision Tree (DT), and Artificial Neural Network (ANN) were developed and evaluated based on performance metrics. Each model was tuned using grid search and random search. The ANN model outperformed the others, achieving an accuracy of 91%. To enhance generalization, the best ANN model was retrained on the entire dataset.

These findings highlight the importance of machine learning in financial decision-making, providing valuable insights for improving predictive banking analytics.

**Keywords**— Predictive Modeling, Banking, classifiers, grid search, random search

## I. INTRODUCTION

Financial institutions use marketing campaigns to promote their services, including term deposit subscriptions. However, predicting which clients are likely to subscribe remains a challenge due to the diverse financial backgrounds, preferences, and behaviors of customers. Traditional marketing strategies often result in inefficient resource allocation and suboptimal client engagement. Hence, a data-driven approach is essential to improve targeting and conversion rates [1], [2].

This study aims to develop a predictive model to identify potential subscribers based on historical campaign data from a Portuguese banking institution. By leveraging machine learning techniques, we seek to enhance marketing efficiency, reduce costs, and increase the likelihood of client engagement. The implementation of predictive analytics enables institutions to focus their efforts on the most promising prospects, ultimately leading to improved customer satisfaction and profitability [3], [4].

The dataset contains demographic, financial, and campaign-related features, offering a comprehensive view of client profiles and their interactions with past marketing efforts. We conduct extensive exploratory data analysis to uncover meaningful patterns and insights. The dataset is then pre-processed through feature encoding, normalization, and data splitting to ensure robust model performance [5].

To achieve our goal, we evaluate and compare three distinct machine learning models: K-Nearest Neighbors (KNN), Decision Tree, and a simple Artificial Neural Network (ANN). These models are assessed based on key performance metrics, including accuracy, precision, sensitivity, specificity, F1-score, and ROC AUC, to determine the most effective approach for predicting client subscriptions. The best-performing model is then selected for deployment, providing financial institutions with a powerful tool to optimize their marketing strategies [6].

## II. LITERATURE REVIEW

Predicting customer behavior using machine learning has been extensively studied in financial and marketing domains. Several research works have focused on bank marketing datasets to improve targeting efficiency and conversion rates. For instance, Moro et al. [1] applied data mining techniques to bank marketing data and found that decision trees and logistic regression performed well in predicting term deposit subscriptions. Deep learning techniques, such as artificial neural networks (ANNs), have been increasingly used for classification tasks due to their ability to capture complex patterns in data [7]. Other works have explored ensemble methods such as Random Forests and Gradient Boosting Machines (GBMs) to enhance prediction accuracy [8]. This study contributes to the existing literature by comparing traditional machine learning models with ANN to determine the best approach for predicting client subscription.

### A. Traditional Machine Learning Approaches

Moro et al. [1] applied data mining techniques to bank marketing data, finding that decision trees and logistic regression performed well for predicting client subscription to term deposits.

Ghatasheh et al. [9] extended these approaches by employing feature selection methods, such as genetic algorithms, to enhance model performance and interpretability.

### B. Deep Learning Approaches

LeCun et al. [7] introduced the power of artificial neural networks (ANNs) for classification tasks, demonstrating their ability to capture non-linear relationships.

Zhang et al [10] provided an in-depth analysis of deep learning-based forecasting models, discussing various model architectures, practical applications, and their respective advantages and disadvantages. Their review also highlights advanced models such as Transformers, generative adversarial networks (GANs), graph neural networks (GNNs), and deep quantum neural networks (DQNNs), offering insights into their effectiveness in price forecasting tasks.

### C. Ensemble and Hybrid Methods

Patwary et al. [11] examined the performance of ensemble learning algorithms, such as Random Forest and Gradient Boosting, to predict whether a new customer would subscribe to a term deposit. Their research highlighted the effectiveness of these ensemble methods in improving predictive accuracy.

Wei et al. [12] developed sophisticated hybrid models that combine various machine learning techniques to estimate the success rate of bank telemarketing campaigns. Their study demonstrated that these hybrid models can effectively predict client subscription behavior, leading to more efficient marketing strategies.

These studies provide valuable insights into different machine learning techniques and their applicability to predicting bank marketing campaign outcomes. Our research builds on this foundation by evaluating traditional models alongside ANN to determine the most effective approach for client subscription prediction.

## III. METHODOLOGY

### A. Data Description

The dataset used in this study originates from a Portuguese banking institution's direct marketing campaigns. It consists of 45,211 instances and 17 features, including the target variable  $y$ , which indicates whether a client subscribed to a term deposit ('yes' or 'no'). The dataset is well-structured and contains no missing values, ensuring a robust foundation for model development. It was downloaded from the Kaggle platform.

The dataset includes a mix of demographic, financial, and campaign-related features, which provide a comprehensive view of client attributes and interactions.

- **Demographic Features:** These features provide information about client characteristics and background:
  - age: Age of the client (numerical).
  - job: Type of job (categorical, e.g., 'admin.', 'technician', 'blue-collar').

- marital: Marital status (categorical: 'married', 'single', 'divorced').
- education: Level of education (categorical: 'primary', 'secondary', 'tertiary', 'unknown').

- **Financial Features:** These features describe the financial status of the client:
  - balance: Account balance in euros (numerical).
  - default: Indicates whether the client has credit in default (categorical: 'yes' or 'no').
  - housing: Whether the client has a housing loan (categorical: 'yes' or 'no').
  - loan: Whether the client has a personal loan (categorical: 'yes' or 'no').
- **Campaign-Related Features:** These features capture information related to the marketing campaign interactions:
  - contact: Communication type (categorical: 'cellular', 'telephone').
  - day\_of\_week: Day of the week when the client was last contacted (categorical: 'mon', 'tue', 'wed', 'thu', 'fri').
  - duration: Last contact duration in seconds (numerical; important predictor for term deposit subscription).
  - campaign: Number of contacts performed during this campaign (numerical).
- **Target Variable:**
  - $y$ : The outcome variable indicating whether the client subscribed to a term deposit (binary: 'yes' = 1, 'no' = 0).

The dataset is well-balanced with a reasonable mix of categorical and numerical variables.

The duration feature is highly correlated with the target variable; however, it can lead to data leakage since longer calls often result in positive outcomes.

Some categorical variables, such as job and education, contain multiple unique values, requiring encoding for machine learning models.

The dataset does not contain missing values, reducing the need for extensive imputation strategies.

This dataset provides a strong basis for applying machine learning models to predict client subscription behavior, allowing financial institutions to improve their marketing efficiency and decision-making processes.

### B. Data Preprocessing

To ensure the dataset was suitable for machine learning models, the following preprocessing steps were applied:

- **Encoding Categorical Variables:** Since machine learning models require numerical inputs,

categorical features such as job, marital, education, and contact were one-hot encoded. This transformation created binary variables for each category, allowing models to interpret them effectively.

- **Scaling Numerical Features:** Features like age and balance were standardized using Min-Max scaling, ensuring they were within a fixed range (0 to 1). This step improved the performance of distance-based models such as KNN.
- **Target Encoding:** The target variable  $y$  was converted into binary values (1 for 'yes', 0 for 'no'), enabling classification models to process the outcome.
- **Train-Test Split:** The dataset was divided into 80% training data and 20% testing data to evaluate model performance on unseen instances.

### C. Machine Learning Classifiers

Three machine learning models; K-Nearest Neighbors (KNN), Decision Tree (DT), and Artificial Neural Network (ANN) were trained and evaluated for predicting client subscription to term deposits. For each classifier, hyperparameter tuning was performed using both Grid Search [13] and Random Search [14] to optimize performance.

#### *K-Nearest Neighbors*

K-Nearest Neighbors (KNN) is a distance-based machine learning algorithm that classifies instances by identifying the majority class among their nearest neighbors [15], [16]. It is a non-parametric, instance-based learning method that does not make explicit assumptions about the underlying data distribution, making it particularly useful for classification and regression tasks. The algorithm assigns a class label to a given data point based on the most common label among its  $k$  nearest neighbors, where distance is used as a measure of similarity.

In this study, the KNN model was implemented using the scikit-learn library [17], with Euclidean distance chosen as the similarity measure. Euclidean distance is a widely used metric that calculates the straight-line distance between two points in a multidimensional space. Other distance metrics, such as Manhattan or Minkowski distance, could be explored to assess their impact on model performance.

To enhance predictive accuracy, hyperparameter tuning was conducted using both Grid Search and Random Search techniques. These methods systematically explore different values of  $k$  (the number of neighbors) and other relevant parameters to identify the optimal configuration. After extensive evaluation,  $n\_neighbors = 13$  was determined to be the best parameter, providing a balance between bias and variance. A lower  $k$  value increases sensitivity to noise, while a higher  $k$  value smooths decision boundaries but may reduce model flexibility.

KNN offers simplicity and effectiveness, particularly for smaller datasets where computational cost is manageable. It is easy to interpret and does not require an explicit training phase, as predictions are made based on stored training instances. However, KNN can be computationally expensive for large datasets due to the need to compute distances for

every new prediction. Additionally, the algorithm can be sensitive to irrelevant or redundant features, making feature selection and data preprocessing critical for optimal performance.

Despite its limitations, KNN remains a valuable tool in classification tasks, particularly when dealing with well-structured and small-to-medium-sized datasets. Future research could explore techniques such as approximate nearest neighbors or dimensionality reduction methods like Principal Component Analysis (PCA) to improve KNN's efficiency in high-dimensional data spaces.

#### *Decision Tree*

Decision Trees are rule-based classifiers that recursively split data based on feature importance to achieve optimal classification [18]. These models operate by creating a hierarchical structure of decision nodes, where each split is determined by the most informative feature that minimizes impurity. The goal is to create pure subsets where the target variable becomes more homogeneous as the tree deepens. Decision Trees are widely used for classification and regression tasks due to their simplicity, interpretability, and ability to handle both numerical and categorical features.

In this study, the Decision Tree model was implemented using the DecisionTreeClassifier from the scikit-learn library. To enhance performance and mitigate overfitting, hyperparameter tuning was conducted using both Grid Search and Random Search. The tuning process focused on optimizing key parameters such as:

- **max\_depth:** Controls the depth of the tree to prevent excessive complexity and overfitting.
- **min\_samples\_split:** Specifies the minimum number of samples required to split an internal node, helping regulate tree growth.
- **criterion:** Determines the function used to measure the quality of a split, with Gini impurity chosen for this implementation as it evaluates how often a randomly chosen element would be incorrectly classified.

Following extensive tuning, the optimal parameters identified were **max\_depth = 36**, **min\_samples\_split = 19**, and **criterion = 'gini'**. These values ensured a balanced trade-off between model complexity and generalization, preventing excessive branching while maintaining predictive accuracy.

Decision Trees provide several advantages, including ease of interpretation, minimal data preprocessing requirements, and the capability to model non-linear relationships. However, they are highly susceptible to overfitting, especially when the tree grows too deep and captures noise instead of underlying patterns. To mitigate this, techniques such as pruning (removing less significant branches), feature selection, and ensemble methods like Random Forest or Gradient Boosting can be employed to enhance model robustness.

Despite their limitations, Decision Trees remain a fundamental component in machine learning, often serving as the building blocks for more complex ensemble models. Future work could explore hybrid approaches that combine

Decision Trees with other algorithms to further improve performance and stability in predictive modeling tasks.

#### *Artificial Neural Network*

Artificial Neural Networks (ANNs) are models capable of learning complex patterns from data [19]. They consist of interconnected layers of neurons that process input data through weighted connections and activation functions, allowing them to learn representations automatically. ANNs are particularly effective in handling large datasets, capturing intricate relationships, and making high-accuracy predictions across various domains.

In this study, a feedforward neural network with a single hidden layer was implemented using the `MLPClassifier` from the `scikit-learn` library. This type of ANN is composed of an input layer, one hidden layer, and an output layer, where information flows in one direction from input to output. The hidden layer enables the network to model complex relationships that linear models may fail to capture.

To enhance model performance, hyperparameter tuning was conducted using Randomized Search, an optimization technique that efficiently explores a wide range of hyperparameter values. The key hyperparameters tuned were:

- **max\_iter**: Defines the maximum number of training iterations to ensure convergence.
- **hidden\_layer\_sizes**: Specifies the number of neurons in the hidden layer, which impacts the model's capacity to learn features.
- **activation**: Determines the activation function used in the hidden layers, which introduces non-linearity to the model and enables it to capture complex patterns.

Following extensive tuning, the optimal parameters found were **max\_iter = 100**, **hidden\_layer\_sizes = 27**, and **activation = 'relu'**. The **ReLU (Rectified Linear Unit) activation function** was chosen for its efficiency in mitigating the vanishing gradient problem and accelerating convergence during training.

ANNs offer significant advantages in predictive modeling due to their ability to model non-linear relationships, adapt to large and complex datasets, and generalize well to unseen data. However, they come with challenges, including high computational requirements and a lack of interpretability. Unlike decision trees, which provide clear decision rules, ANNs function as "black boxes," making it difficult to understand the reasoning behind their predictions.

Despite these challenges, ANNs remain a crucial tool in machine learning applications, particularly when dealing with high-dimensional and unstructured data. Future work could explore deep learning architectures, additional hidden layers, and advanced optimization techniques, such as dropout regularization and adaptive learning rates, to further enhance performance and interpretability.

## IV. RESULTS AND DISCUSSION

This section focuses on the outcomes and discussion of the applied machine learning methods. The performance

metrics, including accuracy, precision, recall (sensitivity), specificity, and F1-score, are evaluated. The k-nearest neighbors (KNN), decision tree (DT), and artificial neural network (ANN) models are analyzed.

The working environment is Python, a robust language well-suited for scripting and rapid application development across various fields and platforms. We opted for Jupyter Notebook, an interactive computational environment, for implementing and testing the models. The computations were conducted on a personal computer equipped with an Intel Core i5 2.60 GHz CPU and 12 GB of RAM, running Windows 10.

Table 1 presents a comparative analysis of the performance of three machine learning models, Artificial Neural Network (ANN), K-Nearest Neighbors (KNN), and Decision Tree (DT) in predicting client subscription to term deposits. This evaluation aims to highlight the strengths and weaknesses of each model in terms of predictive capabilities, considering various performance metrics such as accuracy, recall, precision, F1-score, and specificity.

#### *Accuracy Analysis*

Accuracy serves as an initial measure of a model's overall correctness in classification. In this case, all three models exhibit relatively high accuracy, with ANN achieving 0.91, DT following closely at 0.90, and KNN slightly behind at 0.89. These values suggest that all models perform well in distinguishing between subscribing and non-subscribing clients. However, accuracy alone is not a sufficient metric, particularly in imbalanced classification problems, as it does not reflect how well a model identifies actual subscribers. A model may achieve high accuracy simply by favoring the majority class, which can be misleading in assessing real-world applicability.

#### *Recall: Identifying Subscribers*

Recall is crucial in applications where it is important to correctly identify as many actual subscribers as possible. ANN outperforms the other models in this regard, achieving a recall of 0.48. This indicates that ANN is the most effective at capturing clients who genuinely subscribe to term deposits. DT follows with a recall of 0.43, while KNN struggles significantly, registering a recall of only 0.17. The low recall score of KNN suggests that it fails to detect a large portion of actual subscribers, making it a suboptimal choice for applications where maximizing true positives is essential, such as targeted marketing campaigns focused on potential customers.

#### *Precision: Avoiding False Positives*

Precision, on the other hand, measures how often a model's positive predictions are correct. Here, KNN performs the best, achieving a precision of 0.73. This means that when KNN predicts a client will subscribe, it is correct 73% of the time, making it the least prone to false positives. ANN and DT exhibit slightly lower precision scores of 0.65 and 0.62, respectively. While ANN excels at identifying true subscribers (as seen in its recall score), its lower precision suggests that it also generates a higher number of false positives compared to KNN. This trade-off is important to consider, particularly in business contexts where false positives could lead to wasted marketing efforts and increased costs.

### F1-Score: Balancing Precision and Recall

The F1-score provides a balanced measure that takes both precision and recall into account. ANN achieves the highest F1-score at 0.56, indicating that it strikes the best balance between capturing actual subscribers and minimizing false positives. DT follows with an F1-score of 0.50, while KNN has the lowest score at 0.27. The poor F1-score of KNN confirms that, despite its strong precision, its extremely low recall makes it an unreliable choice for applications requiring a balance between false positives and false negatives. In contrast, ANN's higher F1-score suggests that it is the most robust model when both recall and precision are considered equally important.

### Specificity: Identifying Non-Subscribers

Specificity measures how well a model correctly identifies non-subscribers. In this regard, KNN (0.99) and DT (0.98) perform exceptionally well, making them highly effective at avoiding false positives. ANN, while still strong at 0.96, falls slightly behind. High specificity is valuable in marketing scenarios where the goal is to minimize outreach to clients who are unlikely to subscribe, thereby improving the efficiency of targeted marketing efforts.

The comparative performance of the three models is further illustrated in the accompanying graph. The plotted curves allow for a visual interpretation of how each model performs across different metrics. The ANN curve shows a well-balanced distribution between recall and precision, reinforcing its suitability for predictive modeling. In contrast, KNN's curve highlights its strong precision but severely low recall, making it a risky choice for applications that prioritize identifying true subscribers. DT follows a similar trend to ANN but with slightly lower scores, suggesting it is a viable alternative but not the most optimal choice.

The graphical representation presented by figure 1 also helps identify trade-offs between different performance metrics. For instance, KNN's high specificity is evident in the graph, indicating its effectiveness at correctly classifying non-subscribers while struggling with actual subscriber identification. This visual aid supports the numerical analysis by providing an intuitive understanding of model strengths and weaknesses.

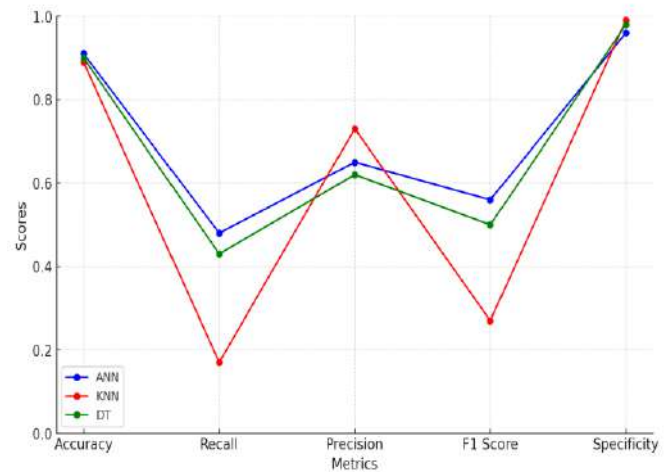
TABLE I. PERFORMANCE COMPARISON OF MACHINE LEARNING CLASSIFIERS

Model	Accuracy	Recall	Precision	F1 Score	Specificity
ANN	0.91	0.48	0.65	0.56	0.96
KNN	0.89	0.17	0.73	0.27	0.99
DT	0.90	0.43	0.62	0.50	0.98

Overall, ANN emerges as the most well-balanced model, offering a favorable trade-off between recall and precision. While KNN exhibits the highest precision, its poor recall significantly limits its effectiveness in identifying actual subscribers. DT performs moderately well across all metrics but does not outperform ANN in any key area. Given its superior F1-score and competitive recall, ANN is the most suitable choice for applications where both capturing true subscribers and minimizing false positives are critical

considerations. This makes it an ideal candidate for predictive modeling in financial services and marketing strategies aimed at maximizing customer engagement with term deposit offers.

FIG 1. MODEL PERFORMANCE COMPARISON



## V. CONCLUSION

This study demonstrated the effectiveness of machine learning models in predicting client subscriptions to term deposits, highlighting their potential for improving financial decision-making. The dataset, comprising 45,211 instances with 17 features, included demographic, financial, and campaign-related variables, providing a comprehensive foundation for predictive modeling. A robust data preprocessing pipeline was implemented, which involved encoding categorical features, scaling numerical features, and splitting the dataset into training and testing sets to ensure reliable model evaluation.

Three machine learning models, K-Nearest Neighbors (KNN), Decision Tree (DT), and Artificial Neural Network (ANN) were developed and assessed based on performance metrics. Among them, the ANN model exhibited the highest predictive performance, achieving an accuracy of 91%. This result underscores the strength of deep learning techniques in capturing complex patterns within financial data. To further enhance model performance, hyperparameter optimization was conducted using both Grid Search and Random Search, refining key parameters to improve accuracy and generalization. After identifying the best-performing ANN model, it was retrained on the entire dataset before deployment to maximize its predictive capability.

These findings highlight the significant role of machine learning in banking and financial analytics, enabling institutions to optimize marketing strategies and enhance customer targeting. By accurately predicting client subscription behavior, banks can allocate resources more efficiently, personalize marketing campaigns, and improve overall customer engagement. Moreover, machine learning models provide a data-driven approach to decision-making, reducing reliance on traditional heuristic methods.

While this study successfully demonstrated the predictive power of machine learning, there are several avenues for future research. Advanced neural network architectures, such as deep learning models incorporating convolutional or

recurrent layers, could be explored to capture even more intricate relationships within the data. Additionally, improved feature engineering techniques, such as automated feature selection and dimensionality reduction, could enhance model interpretability and efficiency. Hybrid machine learning approaches that combine multiple algorithms may also improve overall predictive accuracy. Furthermore, integrating real-time data streams and explainable AI techniques could provide financial institutions with more transparent and dynamic decision-support systems.

In conclusion, this study reinforces the value of machine learning in financial analytics and predictive banking. As technology advances, the integration of AI-driven models into banking operations will become increasingly critical, enabling institutions to stay competitive and better serve their clients. Future research and innovations in this field will continue to refine predictive models, paving the way for more intelligent and data-driven financial strategies.

## REFERENCES

- [1] S. Moro, R. M. S. Laureano, et P. Cortez, « Using data mining for bank direct marketing: an application of the CRISP-DM methodology », *Expert Syst. Appl.*, vol. 37, n° 6, p. 5122-5128, oct. 2011.
- [2] W. Verbeke, D. Martens, C. Mues, et B. Baesens, « Building comprehensible customer churn prediction models with advanced rule induction techniques », *Expert Syst. Appl.*, vol. 39, n° 3, p. 4726-4736.
- [3] T. A. Abdulsalam et R. Tajudeen, « Artificial Intelligence (AI) in the Banking Industry: A Review of Service Areas and Customer Service Journeys in Emerging Economies », *Bus. Manag. Compass*, vol. 68, p. 19-43, sept. 2024, doi: 10.56065/9hfvqrq20.
- [4] C.-F. Tsai et Y.-H. Lu, « Customer churn prediction by hybrid neural networks », *Expert Syst. Appl.*, vol. 36, n° 10, p. 12547-12553, déc. 2009, doi: 10.1016/j.eswa.2009.05.032.
- [5] A. Lemmens et C. Croux, « Bagging and Boosting Classification Trees to Predict Churn », *J. Mark. Res.*, vol. 43, n° 2, p. 276-286, mai 2006, doi: 10.1509/jmkr.43.2.276.
- [6] P. Cortez et M. J. Embrechts, « Using sensitivity analysis and visualization techniques to open black box data mining models », *Inf. Sci.*, vol. 225, p. 1-17, mars 2013, doi: 10.1016/j.ins.2012.10.039.
- [7] Y. LeCun, Y. Bengio, et G. Hinton, « Deep learning », *Nature*, vol. 521, n° 7553, p. 436-444, mai 2015, doi: 10.1038/nature14539.
- [8] R. Caruana et A. Niculescu-Mizil, « An empirical comparison of supervised learning algorithms », in *Proceedings of the 23rd international conference on Machine learning - ICML '06*, Pittsburgh, Pennsylvania: ACM Press, 2006, p. 161-168. doi: 10.1145/1143844.1143865.
- [9] N. Ghatasheh, I. Altaharwa, et K. Aldebei, « Modeling the Telemarketing Process using Genetic Algorithms and Extreme Boosting: Feature Selection and Cost-Sensitive Analytical Approach », *IEEE Access*, vol. 11, p. 67806-67824, 2023, doi: 10.1109/ACCESS.2023.3292840.
- [10] C. Zhang, N. N. A. Sjarif, et R. Ibrahim, « Deep learning models for price forecasting of financial time series: A review of recent advancements: 2020–2022 », *WIREs Data Min. Knowl. Discov.*, vol. 14, n° 1, p. e1519, 2024, doi: 10.1002/widm.1519.
- [11] M. J. A. Patwary, S. Akter, Md. S. Alam, et A. N. M. Rezaul Karim, « Bank Deposit Prediction Using Ensemble Learning », vol. 2, p. 42-51, août 2021, doi: 10.37256/aie.222021880.
- [12] W. Guo, Y. Yao, L. Liu, et T. Shen, « A novel ensemble approach for estimating the competency of bank telemarketing », *Sci. Rep.*, vol. 13, n° 1, p. 20819, nov. 2023, doi: 10.1038/s41598-023-47177-7.
- [13] J. Bergstra, R. Bardenet, Y. Bengio, et B. Kégl, « Algorithms for Hyper-Parameter Optimization », in *Advances in Neural Information Processing Systems*, Curran Associates, Inc., 2011. [https://papers.nips.cc/paper\\_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html](https://papers.nips.cc/paper_files/paper/2011/hash/86e8f7ab32cfd12577bc2619bc635690-Abstract.html)
- [14] J. Bergstra et Y. Bengio, « Random search for hyper-parameter optimization », *J. Mach. Learn. Res.*, vol. 13, n° null, p. 281-305, févr. 2012.
- [15] G. Guo, H. Wang, D. Bell, Y. Bi, et K. Greer, « KNN Model-Based Approach in Classification », in *On The Move to Meaningful Internet Systems 2003: CoopIS, DOA, and ODBASE*, vol. 2888, R. Meersman, Z. Tari, et D. C. Schmidt, Éd., in Lecture Notes in Computer Science, vol. 2888, Berlin, Heidelberg: Springer Berlin Heidelberg, 2003, p. 986-996. doi: 10.1007/978-3-540-39964-3\_62.
- [16] Z. Zhang, « Introduction to machine learning: k-nearest neighbors », *Ann. Transl. Med.*, vol. 4, n° 11, Art. n° 11, juin 2016, doi: 10.21037/atm.2016.03.37.
- [17] F. Pedregosa et al., « Scikit-learn: Machine Learning in Python », *J. Mach. Learn. Res.*, vol. 12, n° 85, p. 2825-2830, 2011.
- [18] W.-Y. Loh, « Classification and regression trees », *WIREs Data Min. Knowl. Discov.*, vol. 1, n° 1, p. 14-23, 2011, doi: 10.1002/widm.8.
- [19] J. Schmidhuber, « Deep learning in neural networks: An overview », *Neural Netw.*, 2014, <https://www.semanticscholar.org/paper/Deep-learning-in-neural-networks%3A-An-overview-Schmidhuber/193edd20cae92c6759c18ce93eeea96afd9528eb>

# Quantum Privacy in Secure Medical Systems

Hamda Slimi<sup>2</sup>, Makhoulf Dourdour<sup>1</sup>, Kahil Moustafa Sadek<sup>1</sup>, Amira Bouamrane<sup>1</sup>, Sahraoui Abdelatif<sup>2</sup>

<sup>1</sup>LIAOA Laboratory, University of Oum el Bouaghi, Oum El Bouaghi 04000, Algeria

<sup>2</sup>University of Tebessa, Tebessa 12000, Algeria

**Abstract**—This paper explores the integration of NIST Post-Quantum Cryptography (PQC) standards and Quantum Key Distribution (QKD) to address growing cybersecurity threats in e-Health systems, particularly against quantum computing attacks. While NIST PQC provides long-term cryptographic security through quantum-resistant algorithms (e.g., Kyber for encryption, Dilithium for signatures), QKD ensures information-theoretically secure key exchange based on quantum mechanics. We present a hybrid security framework in which QKD secures the real-time transmission of sensitive medical data (e.g., remote patient monitoring and IoT device communications) with inherent eavesdropping detection. And NIST PQC safeguards stored health records and authenticates users, protecting against future quantum decryption threats. A case study demonstrates this approach in a telemedicine scenario, showing how QKD-protected channels and PQC-encrypted databases work synergistically to meet GDPR and HIPAA requirements. Key challenges include QKD's distance limitations and PQC's computational overhead, but early implementations in hospital networks prove feasibility. Our analysis concludes that the NIST PQC + QKD combination offers a robust transition path for healthcare systems entering the quantum era, balancing present-day practicality with future-proof security. Further research should optimize interoperability and cost-efficiency for widespread adoption.

**Index Terms**—Post-Quantum Cryptography, Quantum Key Distribution, e-Health Security, NIST Standards, Quantum-Safe Encryption.

## I. INTRODUCTION

The rise of quantum technologies threatens current encryption systems, particularly in the sensitive field of e-health, where the confidentiality of medical data is crucial. According to the National Institute of Standards and Technology (NIST), conventional cryptographic algorithms (RSA, ECC) could be compromised by quantum attacks within 10 to 15 years [1]. At the same time, the growing adoption of telemedicine and connected medical devices (IoT) exposes healthcare institutions to increased risks of cyberattacks [2]. To address this dual problem, two approaches are emerging:

- 1) Post-quantum cryptography (PQC), standardized by NIST since 2022 with algorithms such as Kyber (encryption) and Dilithium (signatures) [3], offers a robust alternative to vulnerable protocols.
- 2) Quantum key distribution (QKD), deployed in real-world networks like the European OpenQKD project [4], guarantees unconditional security for key exchange.

However, few studies explore their synergistic integration into e-health infrastructures, despite critical needs identified by the WHO in 2023 on securing health data [5]. Our article fills this gap by proposing an operational framework combining NIST

PQC and QKD, evaluated via a real case study in a hospital setting.

## II. STATE OF THE ART

The advent of quantum computers (IBM, Google, China) directly threatens the traditional cryptographic protocols used in e-health systems, namely: Shor's algorithm (1994), which allows for rapid factorization of large numbers, making RSA and ECC (Elliptic Curve Cryptography) obsolete within 5 to 10 years according to a NIST report (2022) [6]. And Grover's algorithm, which accelerates brute-force attacks, reduces the effective security of AES-256 to 128 bits [7]. This has a direct impact on e-health, with reports of hacking into patient records stored with vulnerable algorithms (e.g., harvest now, decrypt later attacks). This includes the interception of medical data flows (telemedicine, IoT) via man-in-the-middle attacks. The limitations of traditional cryptographic solutions present critical shortcomings, such as symmetric encryption (AES), which suffers from reduced security against Grover, requiring longer keys (high computational cost). Also, public-key cryptography (RSA, ECC) is completely vulnerable to Shor, compromising digital signatures and key exchanges. E-health suffers from specific problems such as unacceptable latency for medical IoT devices (e.g., pacemakers) with heavy algorithms and difficult interoperability between heterogeneous systems (hospitals, laboratories). As for Emerging Solutions and Their Challenges, two promising, but imperfect, approaches are emerging:

TABLE I  
COMPARISON OF POST-QUANTUM CRYPTOGRAPHY AND QUANTUM KEY DISTRIBUTION

Advantages	Limits
<b>Post-Quantum Cryptography (NIST PQC)</b>	
- Standardized algorithms (Kyber, Dilithium) resistant to Shor/Grover [8].	- Computational overhead: Kyber requires twice as much resources as RSA (study by Bernstein et al., 2023) [9].
- Compatibility with existing infrastructures (software updates).	- Limited trust: Risk of undetected vulnerabilities (e.g., lateral attacks).
<b>Quantum Key Distribution (QKD)</b>	
- Unconditional security based on the laws of quantum physics (BB84 protocol).	- Maximum distance (~100 km in optical fiber, requiring expensive quantum repeaters).
- Real-world deployments (e.g., SECOQC network in Europe, Micius project in China) [10].	- Photon fragility: Sensitivity to environmental disturbances (optical noise).

There are few studies in the literature on the integration of NIST PQC and QKD for e-Health (due to the need for

hybrid frameworks). Thus, there is a lack of benchmarks comparing performance and security in real-world medical scenarios. While quantum solutions (PQC, QKD) offer viable avenues, their practical deployment in e-Health requires overcoming technical and economic challenges. Our work proposes a hybrid architecture that combines their advantages while minimizing their limitations.

### III. CONCLUSION

eHealth is a critical sector where data protection must be infallible. By combining QKD for sensitive transmissions (medical alerts, IoT) and NIST PQC for long-term storage (patient records), healthcare organizations can anticipate quantum threats while ensuring immediate confidentiality. eHealth (electronic medical records, telemedicine, medical IoT) requires extreme data protection (GDPR, medical confidentiality) in the face of cyberthreats and the future power of quantum computers. The flaws in traditional systems (RSA, AES) and the quantum threat require innovative solutions, namely the use of NIST PQC for robust data encryption and the use of QKD for secure transmission of sensitive information. NIST PQC (post-quantum cryptography) and QKD (quantum key distribution) offer a complementary solution: NIST PQC standardizes algorithms (Kyber, McEliece) that are resistant to quantum attacks, securing the storage of medical records and authentication. And QKD uses the laws of quantum physics for tamper-proof transmissions (BB84 protocol), ideal for telemedicine and medical IoT devices. A practical scenario demonstrates their synergy: a hospital uses QKD to exchange sensor data in real time, while NIST PQC encrypts the archives. This hybrid approach guarantees immediate confidentiality (QKD detects any interception) and long-term resistance (NIST PQC against future quantum attacks). The challenges remain the cost of QKD and the gradual adoption of NIST standards, but these technologies position eHealth as a pioneer in quantum cybersecurity.

### REFERENCES

- [1] National Institute of Standards and Technology (NIST). "Post-Quantum Cryptography Standards." NIST, 2024. <https://csrc.nist.gov/projects/post-quantum-cryptography>.
- [2] ENISA. (2023). Threat Landscape for the Healthcare Sector.
- [3] Jiang, H., Ma, Z., Zhang, Z. (2023). Post-quantum Security of Key Encapsulation Mechanism Against CCA Attacks with a Single Decapsulation Query. In: Guo, J., Steinfeld, R. (eds) Advances in Cryptology – ASIACRYPT 2023. ASIACRYPT 2023. Lecture Notes in Computer Science, vol 14441. Springer, Singapore. [https://doi.org/10.1007/978-981-99-8730-6\\_14](https://doi.org/10.1007/978-981-99-8730-6_14).
- [4] Brauer M, Vicente RJ, Buruaga JS, Méndez RB, Braun R-P, Geitz M, Rydlichowski P, Brunner HH, Fung F, Peev M, et al. Linking QKD Testbeds across Europe. *Entropy*. 2024; 26(2):123. <https://doi.org/10.3390/e26020123>.
- [5] OMS. (2023). Digital Health Security Guidelines.
- [6] NIST. (2022). Ravishankar Chamarajnar, Post-Quantum Cryptography: Preparing for a Quantum Future. April 9, 2025.
- [7] Grover, Lov K. "A fast quantum mechanical algorithm for database search." Proceedings of the twenty-eighth annual ACM symposium on Theory of computing. 1996.
- [8] J. Bos et al., "CRYSTALS - Kyber: A CCA-Secure Module-Lattice-Based KEM," 2018 IEEE European Symposium on Security and Privacy (EuroS&P), London, UK, 2018, pp. 353-367. doi:10.1109/EuroSP.2018.00032.
- [9] Yaser Baseri, Vikas Chouhan, Ali Ghorbani, Aaron Chow, Evaluation framework for quantum security risk assessment: A comprehensive strategy for quantum-safe transition, *Computers & Security*, Volume 150, 2025. <https://doi.org/10.1016/j.cose.2024.104272>.
- [10] Cherry Mangla, Shalli Rani, Henry Kwame Atiglah Academic. Secure Data Transmission Using Quantum Cryptography in Fog Computing. *Wireless Communications and Mobile Computing*, Volume 2022. <https://doi.org/10.1155/2022/3426811>.

# Real-Time Image Processing Algorithms for Embedded Systems

BOUFAIDA SOUNDES OUMAIMA<sup>1</sup>, BENMACHICHE ABDEMADJID<sup>1</sup>, MAATALLAH MAJDA<sup>1</sup>

<sup>1</sup>Faculty of Technology, Department of Computer Science, University of Chadli Bendjedid El Tarf, Algeria

Email: s.boufaida@univ-eltarf.dz, benmachiche-abdelmadjid@univ-eltarf.dz, maatallah-majda@univ-eltarf.dz

**Abstract**—Embedded vision systems need efficient and robust image processing algorithms to perform real-time, with resource-constrained hardware. This research investigates image processing algorithms, specifically edge detection, corner detection, and blob detection, that are implemented on embedded processors, including DSPs and FPGAs. To address latency, accuracy and power consumption noted in the image processing literature, optimized algorithm architectures and quantization techniques are employed. In addition, optimal techniques for inter-frame redundancy removal and adaptive frame averaging are used to improve throughput with reasonable image quality. Simulations and hardware trials of the proposed approaches show marked improvements in the speed and energy efficiency of processing as compared to conventional implementations. The advances of this research facilitate a path for scalable and inexpensive embedded imaging systems for the automotive, surveillance, and robotics sectors, and underscore the benefit of co-designing algorithms and hardware architectures for practical real-time embedded vision applications.

**Index Terms**—Real-Time Image Processing, Embedded Systems, Edge Detection, Frame Averaging, FPGA Implementation, Algorithm Optimization, Convolutional Neural Networks (CNN)

## I. INTRODUCTION

This essay presents a comparative study of some image processing algorithms in embedded vision systems which have been benchmarked using real-time recording and artificial image datasets. Low-density parity check and motion vector algorithms detect inter-frame redundancy in every processed field to achieve high frame-rate coded output. Intra-frame prediction filters block-derivable DC coefficients with a residual correction of 2D frequency components to reduce Coded Video Discussion data without compromising picture quality. Intra-field coding, frame-averaging filtering combined with an inter-field alternate macro-block transfer code improves transmission frame rate while reducing coding latency.

Design guidelines are derived to deploy the image processing tasks on a multi-field basis while maintaining data and picture quality robustness. Timing budget gives a better estimation of the maximum number of admissible fields which can be processed in PLDC algorithm, making it possible to dimension the PLDC device accordingly. Picture analysis gives an optimal combination of the other algorithms, achieving ratio results for transmitted frames in mutual cooperation. It is shown that those algorithms can be used independently without compromising their function. Subsequent coded video presentation gives a comprehensive and illustrative view of

the video sequences before and after being processed by these algorithms. Fixed point simulation analysis of the deployed algorithm in the target architecture shows that picture quality robustness is preserved.

Embedded Vision Systems capturing, treating and transmitting images is a critical function in the automotive domain. Pioneer work pioneered it by formulating image compression, transmission and graphical user interface (GUI) rendering in the standard C programming language. Previous works developed versatile image processing algorithms predicting inter-frame redundancy, basing the treatment on integral z-transform. Some algorithms were adapted to hardware implementation on a Xilinx Virtex-II field programmable gate array (FPGA) with embedded microprocessor. This ensured in real time processing compliance with the ISO 26262 hardware safety integrity level (ASIL) requirements. Most of these algorithms were developed considering a single frame basis and/or could not run complementary to each other.

## II. BACKGROUND OF AND MOTIVATION

The invention of embedded systems or computers has changed the world completely. All the devices around us are either embedded systems or they contain embedded systems. The word embedded system refers to a dedicated computer system that is designed for a particular task. They are used in many applications in defense, health care, automobile, etc. The size of the embedded systems has been reduced to a size that can be integrated into any device on the earth. Many applications give high importance to real time processing because real time processing has to follow certain deadlines. There are many applications in which the processing is done according to user choice but in few applications, processing has to be done in real time according to certain conditions. There are many applications in which the processing is done on signal sampling basis and the applications are called as sampling systems. The time taken to process each sample is called period and the time between two consecutive samples taken is called sampling interval. If the signal to be processed is continuous with respect to time then it is called analog signal or continuous signal. The continuous time signal is converted into discrete time signal at regular intervals and the process of converting continuous time signal to discrete time signal is called as sampling. The cost of the sampling system will increase with increase in number of processes in it. As the number of conditions or parameters it takes care of increases

the complexity of the system increases. The processing of the single sampled signal with consideration of certain parameters is called as single conditions processing system. These systems can be easily implemented with the help of embedded systems containing digital signal processors or microcontrollers. Image processing systems are the type of signal processing systems in which the information is in the form of image. The images will be two dimensional and image processing systems will process the images of pixels. There are many image processing systems in which images are analysed and the information is obtained. An embedded based image processing system will analyse the different features of the image taken at a particular time. The images can be seen by humans or they can be sensed with the help of cameras or other devices. Real time image processing systems are the systems which check for particular conditions at regular time intervals. If there is any deviation with respect to the condition then appropriate action will be taken. Real time image processing systems can be used for detecting vices or monitoring places like banks, ATM centers, Entry and exit of sensitive places like laboratories, etc [1]. There are many embedded systems which process the image in the digital or analog form but are not capable of processing in real time. There is a need of embedded systems with image processing capable of processing the images in real time.

### III. SCOPE AND OBJECTIVES

The Scope and Objectives of the essay are outlined in this section. It provides clarity on the boundaries within which the topic is explored and what specific goals are aimed to be achieved.

#### A. Scope

Monitoring of surveillance places has become a need recently, and this had to be fulfilled with other embedded image processing systems. Also, the images captured in these systems should be processed in real-time for efficient performance. There are various algorithms that are designed and standardized for image processing. The various image processing algorithms such as edge detection, object recognition, etc., are implemented here using the C81xx processor (525 MHz) from Texas Instruments [2]. This processor has many features, which provide dedicated DSP features as well as video enhancement features suitable for image processing. A visual board is designed around this processor with all required peripherals such as display and camera to achieve the project objectives. The performance of the different algorithms is compared with the fixed point and floating-point processing.

#### B. Objectives

This section will serve as an introduction to the product. The intention is to stimulate interest in it and give an overview of its most important features [1]. More detailed information will be supplied in the body of the report, including application contexts and background theories. The image processing is one of the efficient tools for enhancing the captured image.

There are many image enhancement techniques available for enhancing and processing the images. The latest DSP processors are being used to overcome the difficulty of processing time and arrangement of the hardware. Addition and difference of the images, various filters, morphological operations, data compression schemes, etc. are most widely used techniques in the existing image processing systems.

### IV. FUNDAMENTALS OF REAL-TIME IMAGE PROCESSING

This section serves as a foundation for understanding real-time image processing. Basic concepts and principles essential for comprehending subsequently advanced topics of this research area are covered. The first half of the section provides an overview of the basic concepts of images, digital images and images in their mathematical representation. The second half deals with basic image processing concepts like filtering, enhancement and reconstruction. The real-time image processing basics presented in this section provide a good foundation to understand the complex topics of vision algorithms like stereo vision, optical flow, motion tracking, object recognition, pattern matching, active vision etc [3].

#### A. Image Processing Basics

An image is a two-dimensional signal (or three-dimensional if colors are included) received from some physical scene [4]. An image may also be described as a two-dimensional function  $f(x, y)$ , where  $x$  and  $y$  are spatial (plane) coordinates, and the amplitude of  $f$  at any pair of coordinates  $(x, y)$  is called the intensity or gray level of the image at that point. An image can be viewed both in spatial and frequency domains. In the spatial domain, images are expressed as functions of 2D space co-ordinates  $x$  and  $y$ . In the frequency domain, an image is expressed in terms of the amplitudes and phases of the sinusoidal signals. Spatial imagery formed through the projection of scenes is studied for improving and extracting the data contained in the images. Typical applications exist in areas such as video surveillance, remote sensing, medical imaging, and non-destructive testing. Image processing is generally grouped into two broad categories: conventional techniques and real-time image processing techniques [5].

The aim of a typical image processing system is either to improve the quality of the image or extract useful information from the image. Some common image improvements include decreasing blurriness, increasing contrast, eliminating noise, and eliminating other artifacts that prevent a good visualization of the image. Such applications include the restoration of medical samples before diagnosis, improving image quality in remote sensing scenarios, and decoding and displaying images received from low-resolution cameras mounted on satellites or Simple Observation Platforms around the earth. A large number of pixels need to be processed in a short time to make the above task viable. Real-time implementation of image processing algorithms is extremely challenging since the hardware resources of typical image processors are very limited.

### B. Real-Time Constraints

Real-time processing is the ability to keep up with the incoming data stream. When discussing real-time, there is often confusion about the different flavors of real-time. There are hard real-time systems that cannot tolerate delay and will cause catastrophic failures, such as flight control systems for aircraft. There are soft real-time systems that may fall behind, but can tolerate some missed frames, such as multimedia streaming. In fact, the terms hard and soft real-time have more to do with business decisions made than with the data stream itself. When discussing real-time processing, it is useful to provide an example of data that may be processed.

The previous sections have focused mostly on image processing. It is common, especially in the scientific community, to be beset by the image. However, images are just one data stream and there are a number of others that can be used and been used effectively. Audio may be considered the second most common type of data. In fact, image processing systems are often hard because video is composed of images with temporal constraints [2]. Both images and audio display periodic behavior. However, unlike images, audio is a one-dimensional data stream. Ultimately, it is hoped to convince that the screen is optional when it comes to processing images of the appropriate data representation. Nevertheless, the focus will remain on images for this section.

Both of these data streams have hard real-time constraints on them. This can be best categorized by describing what happens if the processing does not keep up. For images, they build up in a buffer waiting to be processed, adding lag to the system. In the worst case, the buffer causes system memory to overflow and crash. For audio, if the processing cannot keep up the data is simply discarded. In the worst case, there will be a terrible noise that is unlistenable. Understanding the response time requirements for a data stream is imperative to designing a real-time system capable of processing that data stream [6]. The underlying hardware for anything more complicated than trivial processing will be vastly different depending on the data stream.

## V. FAST FOURIER TRANSFORM (FFT) IN REAL-TIME IMAGE PROCESSING

The DFT relates a sampled function in the time domain to its sampled function in the frequency domain. It is a complex-valued summation requiring  $O(n^2)$  multiplications and additions, with  $n$  the number of samples. It is linear, periodic, and can be viewed in the frequency domain as the transfer function of a filter bank. The fast Fourier transform (FFT) computes the DFT in  $O(n \log n)$  time using the divide-and-conquer paradigm. Formal elegance is not the hallmark of the FFT. This tutorial describes the betraying splendor of the radial butterfly and its cousins.

### A. Overview of FFT

For many applications in communication systems, medical imaging, navigation, process control, remote sensing and in many others, real-time image processing is of great interest.

In such applications, filtering algorithms such as convolution filter or correlation filter are computationally expensive which involves large number of multiplications and additions between the pixels of the image and filter coefficients. Filtering operations involving large number of pixels in the image and large filter coefficients are implemented efficiently in the frequency domain, where filtering operations require fewer number of computations. To convert image from spatial domain to frequency domain and vice versa, Fourier Transform is used [7]. Basic Fourier Transform is highly complex algorithm and has large time complexity which renders it unsuitable for real-time applications and so a Fast Fourier Transform (FFT) algorithm is required. Basically, there exists a trade-off between speed and hardware requirement of FFT algorithms. In general, FFT processors fall under two designs i.e. high-speed design architectures where throughput is increased at the cost of increased hardware resources and scalable architectures where throughput is increased at the cost of reduced hardware resources [8]. Thus, single-path delay feedback (SDF) multi rate FFT processors can be used to address the applications which require high speed with minimal hardware resources.

### B. Applications in Real-Time Image Processing

The two-dimensional discrete Fourier transform (2D DFT) is important for many applications including image processing, pattern recognition and machine vision. Fourier image analysis has been used to reduce computational complexity of many complex convolution operations involving derivative operators in the spatial domain by converting them into simple multiplications in the frequency domain [9]. Nevertheless, as a direct consequence of using DFTs, large amounts of images that are required for processing can lead to bottlenecks in machine vision applications necessitating high-speed parallel architectures. In addition to being memory constrained, DFTs in many systems form bottlenecks in the control loop of the machine vision tasks, making it essential to accelerate their computations. This is particularly the case for standard image sizes such as those from the common CCIR or PAL cameras with a resolution of  $512 \times 512$ . On such images, the computational requirements of 2D DFT are inherently high: 17 msec or longer using 8-bits integer 2D DFT for the TI TMS320C6701 DSP (213 MFLOPS) that is comparable to other state of the art DSPs, C and other processors. It is well known that the Cooley-Tukey Fast Fourier Transform (FFT) algorithm reduces the computational complexity of N-point DFTs from  $O(N^2)$  to  $O(N \log N)$ . However, in the case of 2D DFTs, it has been shown that the implementation of 1D FFTs has to be computed in two-dimensions, i.e., 1D FFT has to be implemented before and after the row-to-column data transposition. This increases the computational complexity of the design, in addition to the control and management circuitry, making 2D DFTs a significant computational bottleneck for many real-time machine vision applications. There exist several resource-efficient, high-throughput implementations of 2D DFTs. However, the majority of the currently available on chip FPGA based 2D FFT FPGA implementations rely on

a repeated invocation of 1D FFTs by the row and column decomposition (RCD) [10]. The RCD schemes have been proposed and hardware architectures described to achieve for the most commonly used PPUs, fully pipelined with parallel processing and a number of very promising systems achieving real-time or near real-time performance upon the standard image such as the previous examples. One significant complexity issue/challenge with RCD schemes is the transposition switch which is often very difficult to design efficiently due to the several complex back-to-back multiplexers. In practice, all RCD-based designs either partially or entirely avoid an all-to-all data transposition.

## VI. CONVOLUTIONAL NEURAL NETWORKS (CNNs) IN REAL-TIME IMAGE PROCESSING

Convolutional Neural Networks (CNNs) are composed of layers, where each layer consists of a number of filters (also referred to as convolutional kernels). These filters are trained during the training phase and are employed by the network to perform image processing operations like edge detection, texture filtering and so on [11]. CNNs can be employed to classify an image at a broader level (semantic labels of the image), while models known as Fully Convolutional Networks (FCNs) carry out pixel classification tasks (instance level labeling). CNNs and FCNs have been shown to yield state-of-the-art performance in various high-level vision applications such as object detection and recognition in images and videos, semantic segmentation, and artistic style transfer.

Implementing computer vision algorithms based on CNNs and/or FCNs poses critical challenges for Embedded Systems, which need to simultaneously ensure real-time processing and power consumption at a given threshold. Two broadly defined challenges can be identified here: 1) Network complexity and resource consumption, and 2) Input image complexity. In benchmark datasets like ImageNet, videos to be processed have a resolution of  $227 \times 227$  pixels and only three-color channels. Since cameras in Embedded Systems usually output high-resolution color images (e.g.,  $> 1$  megapixel), there is thus a compelling need for optimizing neural networks tackling image processing tasks. Therefore, real-time video/image segmentation and processing with high-resolution images on embedded platforms remain an open challenge [12].

### A. Introduction to CNNs

CNNs are an extensively employed deep learning architecture, which allows the automatic extraction of multi-scale image features without requiring prior knowledge to be integrated by users. A CNN has four main building blocks, including a convolution layer, a pooling layer, a fully connected layer, and an output classifier [13]. A convolution layer has  $M$  filters, which are trained offline and passed to the architecture. Each filter is a 3D volume with  $(3 \times 3 \times K)$  weights (three for each RGB channel) and one bias. The parameters in the convolution layer are optimized offline by back-propagation using the stochastic gradient descent optimization method involving labelled images. The result of a convolution is a joint

convolved image, which indicates how strongly each filter at a certain position responds to the input image. Pooling layers are aimed at reducing the dimensionality and robustness to allowable image deformations. Therefore, the scale-invariance property of pooling layers reduces the representation size while maintaining the maximum important information. The pooling operation is performed after the convolution between the input image and filter responses, where the pooling size determines how much of shrinking size the operation would have. Each pooling layer can be followed by several convolution layers to decrease the picture size gradually. A fully connected layer performs a matrix multiplication between the vectorized response of the last layer and the trained  $M \times N$  weights, where  $N$  indicates the number of units in the fully connected layer. Each fully connected layer is often sawn by a rectified linear unit (ReLU) activation function to allow non-linearity.

CNN features are a function of the learned layer and are expected to be selective for certain local aspects of the input data. For instance, in the early layers, simple features such as edges can be composed, while in deeper layers, this composition is more complex and can involve features such as eyebrows, eyes, etc. The deeper the layer is, the higher-level feature is, and the more complex the learned patterns are. The estimation of features based on CNNs has been widely adopted in earlier works.

### B. Optimization Techniques for Embedded Systems

Various optimization techniques and strategies, tailored for the deployment of CNNs to embedded platforms are presented in Holistic Optimization of Embedded Computer Vision Systems [14]. This covers solutions specifically addressing challenges of real-time image processing, a central capability for autonomous systems, across vision sensors from programmable and heterogeneous SoCs to low-power FPGAs and imager with processor co-designing. Prototyping, benchmarking and model deployment workflows across neuron abstraction-levels from low-precision fixed-point weights to tiny-LAT-based models and hardware frameworks are also described. These techniques can significantly reduce a CNN's computational complexity and memory requirements. Methods for quantizing a model's weights and activations to 8 bits or lower fixed-point with proven robust performance, forming fast and area-efficient designs for programmable platforms are demonstrated. Memory access requirements can be reduced by converting CNNs into two-stream architectures where activations at computation-levels are reused on-chip and redundant computations are eliminated [15]. Further reductions in area and latency are achieved through the pruning of weights and neurons of the models with respect to pre-trained performance, resulting in designs comparable within reach of hardware framework's RAM budget. To improve model mapping on programmable SoCs and heterogeneous systems with accessible memory hierarchies and IP cores, multi-dimensional scheduling methods are also developed. These algorithms aim to maximize performance on embedded hardware with fixed

performance metrics through the orchestration of data transfer, access pattern and computation.

## VII. PARALLEL COMPUTING TECHNIQUES IN REAL-TIME IMAGE PROCESSING

The ability to parallelly process data has gained importance in modern image processing applications. Multiple processors can be used to manipulate large images, which would otherwise be cumbersome when processed by a single processor. Heavy processing loads are increasingly imposed on different embedded systems as imaging devices with larger resolutions become available. Image acquisition rates in today's cameras are also on the rise. Processing images in real time is key to many applications. This imposes stringent requirements on speed to processors which may deal with large image databases. It has also become necessary to make various algorithms computationally simpler. Parallel processing is a possible solution, which achieves more speed by conducting some operations simultaneously. Nevertheless, the extent to which this would be beneficial for real time processing based on specific applications needs careful analysis.

In order to study the possible gains available from parallel processing in image processing applications, typical algorithms for image filtering and transformation are examined. The degrees of parallelism achievable with these algorithms on ICO-V and other parallel computers are examined. Some strategies to be followed for processing images in parallel are also pointed out. The parallel image processing algorithm is constructed after taking into account the strategies and the architecture of an ICO-V parallel computer. Experimentally attainable speedups for this case differ very much from theoretical speedups. Some performance results are presented. Close to linear speedup can be expected with increasing number of processing elements for image frame filtering when square filter masks are used. However, the degree of parallelism diminishes with increasing filter mask size. The presence of large numbers of pixels with a given intensity may cause the speedup for binary image processing based on decision trees to approach a constant value even in a very highly parallel architecture [6].

### A. GPU Acceleration

As the number of pixels and cameras in an image processing event gets higher, the parallel processing ability of a system becomes vital for ad-hoc real-time performance. In recent years, the use of Graphics Processing Units (GPUs) provides an ability of high computational throughput, capable of processing 1000s to millions of paralleled tasks at once. The massive parallel processing architecture of GPU brings significant advantages for Real-Time Image Processing Applications when there are lots of processing units with simple math operations in an image data. There are several architectures available for embedded systems equipped with GPUs. In this paper, an experimental study is accomplished to compare the performance of OpenCV built-in CPU and GPU functions on a Cortex A8 Nvidia Tegra 2 based embedded

platform. In addition, the performance on a quad-core CPU platform is investigated for better understanding the behavior of processing algorithms on parallel architectures [6].

Managing a vast processing capability is not only dependent on the architecture of the supporting system but also the algorithmic performance on the system is quite essential. In a limited time, frame of an image sensor event, an out-sourced algorithm must be thought fast enough to handle the real-time requirement. The ratio of the time to run a particular process with a particular algorithm, versus the ratio of the time is consumed to run the same process with another algorithm is called algorithmic performance. It is expected to build self-compliant embedded versions of the computationally intensive Image Processing Applications designated to AmpliSens Processor, NVIDIA's low power Quad-core ARM Cortex A57 generation architecture [16].

### B. Parallel Processing on FPGAs

There is a focus on parallel processing for real-time image processing using Field-Programmable Gate Arrays (FPGAs). Increasing demand for high-speed processing results in a need for parallel processing. In the case of image processing applications, it is usually very complex and requires a large number of operations. Hence, a large amount of data is transferred which requires high-speed data transfer. Therefore, multiprocessing or parallel architecture is needed to cope with this situation. Multi-tasking operating systems and bus architectures can provide parallel processing but not beyond a limit because of the serial nature of data transfer. Therefore, some hardware in which multiple processing cores can be used simultaneously is preferred. In this regard, FPGA helps in parallel processing, and hence a better alternative for real-time image processing. FPGA is nothing but pre-fabricated integrated circuits used to build digital circuits. It consists of logic cells, I/O pins, interconnects, and programmable interconnects. The entire FPGA architecture can be implemented from HDL (Hardware Description Language) code and is very flexible and supports programmability. Since FPGAs have a dedicated path for data transfer, they also provide a rapid data rate [3]. In general, FPGAs consist of lookup tables as logic cells, and the polynomial function or truth table is entered in LUT, logic operations can be performed, and storage elements are multiplexers. The architecture allows combinational logic and sequential logic. Each slice has multiplexers for input and output routing, enabling robust interconnections between logic cells. Interconnects between IOs and logic cells are also programmable. Each IO can be configured with phase-lock loops and series resistors, making it flexible to work with high/low-speed interfaces. So, implementation of FPGA for image processing is preferred over other platforms. There are many successful image processing applications like it's hard to imagine an application that needs high-speed processing and can't be applied to DSPs. Prerecorded/streaming video can be easily interfaced with low-cost chipsets [17].

## VIII. PERFORMANCE EVALUATION AND COMPARISON

Performance evaluation is an important consideration when developing real-time image processing algorithms. This consideration influences the design, choice and employed methodologies to achieve the desired performance criterion. On Embedded Systems (ES) design, performance evaluation can be a critical aspect of the overall design as it impacts the architectures of the chosen hardware and/or algorithm implementation. If real-time behaviour is desired, the performance evaluation of algorithms is typically necessary to understand performance limits of a system [6]. The metrics applied to determine performance criteria can take on several names, such as Performance Parameters, Performance Evaluation Metrics or Performance Measures, which characterise criteria used to determine performance, which may influence future design decisions. The approach usually consists of applying A to B to obtain C. A can consist of hardware implementations (e.g. ES architectures), algorithm implementations, architectures of the devised processing system and/or images, videos or other types of data to be processed. B may represent implementation methods, parallelism increasing techniques, algorithmic detail alterations and/or deceleration of processing, such as down-sampling the input. C usually indicates performance criteria, which can be the maximization of one of the following or the minimization of others, based on processing performance (e.g. frames per second), efficiency (e.g. amount of resources used per performance) or stability. The aspect C is also applicable to the other performance evaluation aspect B.

The envisioned real-time image processing algorithms are usually compared against traditional image processing algorithms [17]. Comparisons between real-time Dehazing, Denoising, Inpainting, Math Morphology and Resizing approaches against traditional image processing approaches which have close application to the within-outside pixel whether the value of a pixel is calculated and processed if it is different from that of its neighbors. A set of metrics to measure image digitised distortion may be used (e.g., PSNR, MSE, Maximum Absolute Difference, number of pixels out of tolerance). Performance was using a number of full-scaling images across different scenarios (e.g. land-sea, edge, etc.) and implemented in MATLAB for a knowledge based understanding of mathematical processing.

### A. Metrics for Evaluation

The performance of each defined algorithm is evaluated according to the following metrics:

- Latency and throughput are measured directly from the embedded platforms. Frames per second (FPS) is calculated from the observed processing latency, and throughput is expressed as pixel operations per second (POP/s) [6].
- Accuracy is evaluated according to the error rate for the associated image processing application, namely for corner detection, template matching, and optical flow. The detection accuracy (detection rate) is derived from

the number of correctly detected features with respect to the number of features in the reference images, and the matching accuracy is derived from the matching error, which is computed from the squared Euclidean distance between matched feature descriptors [2]. Performance characteristics are analyzed using speedup and efficiency. Speedup is defined as the ratio of the performance of the faster processor or configuration to that of the slower one (using FPS). Efficiency is defined as the speedup divided by the number of processors or cores used in parallel processing.

### B. Benchmarking Studies

To understand the performance of the systems developed experimentally, a number of benchmark Realtime image processing algorithm implementations were carried out on the systems. The image processing algorithms chosen were widely used in various computer vision applications and widely researched in the image processing literature. Some of these algorithms have relatively simple data flow such as thinning, edge detection and histogram operations. Other algorithms have a more complicated data flow such as mesh grid generation and watershed segmentation which are also more computationally intensive. In addition to the image processing algorithms, the image filtering operation used to generate the input image data sets was also chosen for benchmarking because it involves convolution which has a complicated data dependency and is commonly used to preprocess images for edge detection [3]. As a preliminary analysis the algorithms and image filter kernel files used for the analysis were applied to a number of different input image sizes, 8-bit grey level images only. The image processing operations chosen were formed together to form a common data flow of operations that were simulated progressively one after the other [17]. Timing results were generated for each image processing operation individually and collectively to understand how each operated and additionally how each affected the other. These timings give an indication of the average true processing time required for each image processing operation on the different image sizes. This information can then be further developed in the safety of a simulation environment to understand the capabilities of the processors used in potential application systems.

## IX. CHALLENGES AND FUTURE DIRECTIONS

The field of real-time image processing has witnessed significant developments in recent years; however, it still faces considerable challenges in the design of algorithms and the hardware implementation of solutions and systems. Many attempts are being made to apply image processing and computer vision concepts for implementation on embedded platforms. Today, advents in custom hardware, reconfigurable hardware, high-speed chip designs, low-cost components, power-full microcontrollers and DSP architectures, image data compression solutions, and high-speed interfacing protocols provide every opportunity to design systems and solutions that

can work standalone, require low bandwidths, and consume low power. Applications that require high reliability under unforeseen problems, such as with respect to illumination changes and variations in the scene, deem smart processing on-chip, in a vision system, as opposing to computationally intensive tasks executed by off-chip general-purpose hardware.

With the increasing time and effort required for systems integration, system-on-chips, and hardware-in-the-loop simulations are becoming more and more popular. The road towards product realization typically starts with a reference implementation on a number of readily available platforms, such as EDA software for the high-level modeling of systems, FPGAs for prototyping, and higher-level development frameworks on DSP and application-specific processors. Moreover, rapid bandwidth growth develops new possibilities for connection architectures, allowing to make quick hardware enhancements, e.g., by integrating data busses dedicated to a newly implemented vision module [3]. The main driver in this direction is the fast-growing sector of mobile vision applications that imposes low manufacturing and processing costs in combination with high-performance and real-time-processing demands. Therefore, new and smart solutions that allow not to compromise performance on existing high-end platforms need to be investigated.

#### A. Current Challenges

Real-time image processing applications, such as surveillance, robotics, tracking, and gesture recognition, are gaining popularity due to the proliferation of camera sensors. Embedded systems, which are cost-effective alternatives to standard computers, using a single chip for processing, control, and communication, are the first choice for these applications. The implementation of real-time image processing algorithms is taken up in this study. The suitability of embedded processors for implementing the algorithms of edge detection, corner detection and blob detection is evaluated. The design cycle involves algorithm analysis, algorithm mapping to suitable architecture, designing the architecture and optimization at different levels. The architecture is supported with a micro-controller to monitor and inspect the results [2].

#### B. Future Trends

With the advancement in the computational requirements in different applications, the real-time processing has become a major area of research. Image or video processing being the most computationally heavy processing and having the dimensionality as a critical parameter, need some change in the algorithmic approaches in order to comply with the real-time constraints. The recent works in the arena of the algorithm optimization, architecture selection, implementation architecture optimization, and the growing trend of sharing embedded resources over the cloud, gives a strong foothold for future work in the real-time image processing [3]. The conventional camera followed by the FPGA based processing over the embedded system, the results showed the feasibility and requirement of the linked input-output delay less than a

few microseconds as the input to output latency. The connected component is a very basic processing algorithm used in many image analysis applications, from which the real-time floating windows implementation on the FPGA platform with the supporting parameterization was discussed along with some future research endeavors [10]. The combinations of different methods, such as a cloud-based processing approach with linked embedded architecture or multi-focal camera based computational approach, can be a great way to enhance the feasibility as well as the processing weights. Moreover, the processing application being linked with the robotics or controllers would also add another level of complexity, but with recent developments in the robotic pipeline it has become an interesting research avenue.

### X. CONCLUSION AND CLOSING REMARKS


The objective of the work presented is to develop and analyze a set of algorithms that can be used as a basis for further research, design, and development of real-time image processing algorithms for embedded systems, and providing proof of their functionality through simulation and/or implementation trial tests. Each algorithm has been proven to function real-time for a reasonably sized image using either a fixed-point DSP or an FPGA in VHDL at a mandated clock speed. This means architecture-based design work is enabled to allow these algorithms to be incorporated into cameras and video equipment that fall outside of the mainstream application literature considered by the in-depth designs [1]. There are valid scenarios where this work will be required to meet stringent design deadlines with limited financial resources. It is also proposed that the algorithms themselves will fulfill a niche market and be of commercial interest not only to the manufacturers of the cameras and video equipment but also to the component re-sellers and DSP manufacturer directly supplying to them. It is intended that the commercial viability of the work will be pursued through the creation of a new business [3]. Current work to investigate smarter methods of estimating an edge direction or mask orientation is intended. Such an estimate will allow a decision as to whether to process or not for a particular edge direction. In this way overall processing bandwidth can be dramatically reduced by only processing visually significant edges and therefore saving power consumption and costs. Further energy sparing concepts are also being investigated to include sampling at higher frame rates during active pixels and lower frame rates during sustaining illumination environments. In this way overall power consumption can be reduced at lower costs with no significant loss in functionality. It is intended that these ideas and algorithms will promote the realization of a funding proposal to a commercial manufacturer.

### REFERENCES

- [1] P. Trotta, 'Enhancing Real-time Embedded Image Processing Robustness on Reconfigurable Devices for Critical Applications', 2016, doi: 10.6092/POLITO/PORTO/2641174.
- [2] R. L. Gregg, 'Real-Time Streaming Video and Image Processing on Inexpensive Hardware with Low Latency'.

- [3] D. Bhowmik and K. Appiah, 'Embedded Vision Systems: A Review of the Literature'.
- [4] C. Hartmann, M. Reichenbach, D. Fey, A. Yumatova, and R. German, 'A Holistic Approach for Modeling and Synthesis of Image Processing Applications for Heterogeneous Computing Architectures'.
- [5] B. Desai, M. Paliwal, and K. K. Nagwanshi, 'Study on Image Filtering – Techniques, Algorithm and Applications', Jun. 04, 2022, arXiv: arXiv:2207.06481. doi: 10.48550/arXiv.2207.06481.
- [6] B. Ruf, J. Mohrs, M. Weinmann, S. Hinz, and J. Beyerer, 'ReS2tAC—UAV-Borne Real-Time SGM Stereo Optimized for Embedded ARM and CUDA Devices', 2021.
- [7] P. B. Hansen, 'The Fast Fourier Transform'.
- [8] J. Takala and D. F. Qureshi, 'MUAZAM ALI PIPELINED FAST FOURIER TRANSFORM PROCESSOR'.
- [9] F. Mahmood, M. Toots, L.-G. Öfverstedt, and U. Skoglund, '2D Discrete Fourier Transform with Simultaneous Edge Artifact Removal for Real-Time Applications', Mar. 16, 2016. doi: 10.1109/FPT.2015.7393157.
- [10] S. Saha, 'A brief experience on journey through hardware developments for image processing and it's applications on Cryptography'.
- [11] A. Athar, 'An Overview of Datatype Quantization Techniques for Convolutional Neural Networks', Aug. 22, 2018, arXiv: arXiv:1808.07530. doi: 10.48550/arXiv.1808.07530.
- [12] L. A. Camuñas-Mesa, Y. L. Domínguez-Cordero, A. Linares-Barranco, T. Serrano-Gotarredona, and B. Linares-Barranco, 'A Configurable Event-Driven Convolutional Node with Rate Saturation Mechanism for Modular ConvNet Systems Implementation', *Front. Neurosci.*, vol. 12, 2018.
- [13] Y. Du, L. Du, Y. Li, J. Su, and M.-C. F. Chang, 'A Streaming Accelerator for Deep Convolutional Neural Networks with Image and Feature Decomposition for Resource-limited System Applications'.
- [14] 'Buckler\_cornellgrad\_0058F\_11723.pdf'.
- [15] J. Turner, J. Cano, V. Radu, E. J. Crowley, M. O'Boyle, and A. Storkey, 'Characterising Across-Stack Optimisations for Deep Convolutional Neural Networks', Sep. 19, 2018, arXiv: arXiv:1809.07196. doi: 10.48550/arXiv.1809.07196.
- [16] B. Hangün and Ö. Eyecioğlu, 'Performance Comparison Between OpenCV Built in CPU and GPU Functions on Image Processing Operations', 2017.
- [17] M. Qasaimeh, K. Denolf, J. Lo, K. Vissers, J. Zambreno, and P. H. Jones, 'Comparing Energy Efficiency of CPU, GPU and FPGA Implementations for Vision Kernels', May 31, 2019, arXiv: arXiv:1906.11879. doi: 10.48550/arXiv.1906.11879.


# Real-Time Machine Learning for Embedded Anomaly Detection

1<sup>st</sup> Abdelmadjid Benmachiche 

Department of Computer Science  
LIMA Laboratory  
Chadli Bendjedid University  
El-Tarf, PB 73, 36000, Algeria  
benmachiche-abdelmadjid@univ-eltarf.dz

2<sup>nd</sup> Khadija Rais 

Informatics and Systems (LAMIS)  
Echahid Cheikh Larbi Tebessi University Echahid Cheikh Larbi Tebessi University  
Tebessa, 12002, Algeria  
khadija.rais@univ-tebessa.dz

3<sup>rd</sup> Hamda Slimi 

Informatics and Systems (LAMIS)  
Echahid Cheikh Larbi Tebessi University Echahid Cheikh Larbi Tebessi University  
Tebessa, 12002, Algeria  
slimi.hamda@univ-tebessa.dz

**Abstract**—The spread of a resource-constrained Internet of Things (IoT) environment and embedded devices has put pressure on the real-time detection of anomalies occurring at the edge. This survey presents an overview of machine-learning methods aimed specifically at on-device anomaly detection with extremely strict constraints for latency, memory, and power consumption. Lightweight algorithms such as Isolation Forest, One-Class SVM, recurrent architectures, and statistical techniques are compared here according to the realities of embedded implementation. Our survey brings out significant trade-offs of accuracy and computational efficiency of detection, as well as how hardware constraints end up fundamentally redefining algorithm choice. The survey is completed with a set of practical recommendations on the choice of the algorithm depending on the equipment profiles and new trends in TinyML, which can help close the gap between detection capabilities and embedded reality. The paper serves as a strategic roadmap for engineers deploying anomaly detection in edge environments that are constrained by bandwidth and may be safety-critical.

**Index Terms**—Embedded anomaly detection, IoT, Embedded systems, Isolation Forest, One-Class SVM, Lightweight Neural Networks, Threshold-based methods, TinyML

## I. INTRODUCTION

An immediate demand has arisen with the fast adoption of IoT and embedded systems across the critical infrastructure, both in industrial control systems and in smart cities, for real-time, on-the-edge anomaly detection. Sending raw sensor data to the cloud to be analyzed is challenging because of bandwidth issues, latency, and privacy issues, which demand edge-based intelligence [1]. Here, anomaly detectors need to be extremely efficient on very constrained resources such as small memory (usually less than 100 KB), low-power processors, and hard real-time performance requirements, excluding many traditional machine learning methods.

The past few years have been characterized by runaway research efforts on how to fit anomaly detection methods into these limited settings. This urgency is amplified by the growing vulnerability of IoT environments to cyber-attacks exploiting system weaknesses and user awareness gaps [19]. In a survey by Adhikari et al. scan across the wide field of IoT anomaly detection, but it is often the case that many suggested solutions are not viable in real embedded settings, as they are too computationally consuming [1]. The TinyML movement

has addressed this gap to some degree by implementing models that are very lightweight and runnable on microcontrollers. Antonini et al. showed a completely unsupervised and flexible anomaly detection system running on a low-cost IoT retrofitting kit, which demonstrates the functionality of on-device learning [4].

Moreover, the issue of concept drift, i.e., the definition of what normal behavior is, changes over time, and presents a major difficulty for embedded systems that cannot be updated regularly. A survey by Aparcana-Tasayco et al. points out that most machine learning models are not validated to be resilient to dynamic, real-world scenarios, although they demonstrate high accuracy in their normal benchmarks [5]. Consequently, a major challenge is that embedded anomaly detection should be evaluated using multiple criteria and should not only focus on accuracy but also on latency, memory footprint, and adaptability.

Through this brief survey, we cut through the complexity by offering a feasible, hardware-concerned perspective on the most plausible anomaly detection strategies for embedded systems. We bring out the trade-offs of most interest to the practitioners: what algorithms will run on either a Cortex-M microcontroller or a more powerful edge CPU, how they manage this changing nature of normal operation, and what can be used to fairly benchmark them. Our goal is to offer a strategic roadmap for engineers and other researchers as they find their way in the fast-paced intersection of TinyML, edge AI, and real-time security.

The remainder of this paper is organized as follows. Section II surveys the four dominant families of embedded anomaly detection methods: tree-based models, one-class learning approaches, lightweight neural networks, and statistical or threshold-based techniques. We analyze each in terms of detection performance, memory footprint, latency, and suitability for hardware platforms ranging from microcontrollers to edge CPUs. Section IV discusses key trade-offs and emerging hybrid designs, while Section III identifies critical research gaps, particularly around concept drift, benchmarking standards, and adversarial robustness. Finally, Section V outlines promising directions for future work, and Section VI concludes with practical guidance for deploying anomaly detection in real-

world, resource-constrained environments.

## II. EMBEDDED ANOMALY DETECTION KEY APPROACHES

Anomaly detection on embedded and edge devices must balance detection performance with strict constraints on memory, latency, and power. In recent works, research has converged around four dominant methodological families, each offering distinct trade-offs between model complexity, adaptability, and hardware feasibility.

### A. Tree-Based Methods

The class of ensembles based on trees, specifically Isolation Forest (IF), continues to be one of the most popular anomaly detectors used to date in embedded IoT devices because of their linear time complexity, small memory footprint, and data-scale insensitivity. IF separates anomalies by randomly dividing the features with the knowledge that anomalies take fewer splits in isolation.

Recent literature proves that IF is still among the most efficient tree-related solutions in order to detect anomalies in embedded and IoT systems. Vasiljevic et al. suggested federated variant FLiForest, which can optimize IF on MicroPython-based edge devices, achieving an accuracy of more than 96 percent with a memory consumption of less than 160 KB, and emphasized that it fits well in resource-constrained and privacy-sensitive environments [22]. Another paper by Zahoor et al. [24] compared the IF with One-Class SVM and a hybrid CSAD technique in IoT security application that the IF is very robust and computationally efficient, although slightly worse than OCSVM. Chua et al. have demonstrated the performance of IF to detect web traffic anomalies with high accuracy and precision through structured data preprocessing and feature engineering [13]. Besides, other researchers in [11] gave a comprehensive overview of isolation-based algorithms with low complexity, scalability, and robustness of IF, along with its effective support of streaming and distributed edge conditions.

### B. One-Class Learning Methods

One-class learning models, such as One-Class SVM (OCSVM) and Support Vector Data Description (SVDD), are trained on a small boundary around normal data in a feature space. Although it was too memory-intensive historically to require microcontrollers, more recent computational economies (especially in model compression, linear approximations, and support vector reduction) have made systems using edge CPUs implemented feasible. The methods are best in cases where a normal behavior has a complicated, non-linear structure.

Recent studies have greatly contributed to improving one-class learning techniques of IoT and edge anomaly detection through better detection and computational performance. Katbi et al. suggested an enhanced interpolated Deep SVDD autoencoder with adversarial regularization, which improves the latent hypersphere representation to advance class separation in heterogeneous, high-dimensional data of IoT, exceeding the conventional shallow, as well as deep, baselines [15]. Ayad and his colleagues in another research paper [6] proposed a

lightweight single-class detection system based on an asymmetric stacked autoencoder integrated with a profound neural network with detection rates of over 96% and a high accuracy on IoT intrusion data with low inference time, which are suitable enough to be deployed on a real-time basis. The authors in this paper [20] examined one-class classification of IoT malware with TF-IDF and n-gram feature encoding, which demonstrates that they can use a model only trained with benign traffic and have a very high recall, as well as strong accuracy, which is remarkable in the context of changing threats. A paper by Yang et al. [23] suggested an efficient OCSVM model that incorporates the Nyström method, Gaussian sketching, clustering, and Gaussian mixture models and allows using smaller memory and prediction time at a cost of quality per detector, which makes OCSVM more feasible in large-scale and constrained resource IoT applications.

### C. Lightweight Neural Networks

Small neural networks, including quantized autoencoders, 1D-CNNs, and pruned recurrent networks, are becoming more usable in resource-constrained devices of all sizes due to the TinyML revolution. These models identify anomalous events through reconstruction error or prediction variance, including those temporal dependencies that are typically overlooked by statistical or tree-based models.

The analysis and the current developments in lightweight neural network architectures have made it substantially more feasible to conduct deep anomaly detection on resource-limited edge and IoT devices. Sivapalan et al. proposed ANNet, a hybrid LSTM-MLP architecture, for open-source real-time ECG anomaly detector on wearable IoT sensors with about 97% classification performance, which can significantly save energy through edge-level decision-making and gated wireless communication over ARM Cortex-M [21]. Babalola and his colleagues investigated AI-assisted edge cybersecurity through compact neural networks, including MobileNet, SqueezeNet, and TinyML models, noting that such neural networks are more effectively able to provide real-time, low-latency, anomaly detection services at small memory and power constraints [7]. The authors in [12] proposed MemATr, a memory-augmented lightweight transformer to detect video anomalies, which can attain competitive performance with just a small fraction of the number of parameters as a standard transformer model and latency to run of sub-50 ms on a mobile device. Amin et al. have shown that LSTM-autoencoder-based anomaly detection is practical in a microcontroller device, and quantization and TensorFlow Lite can be used to implement real-time monitoring of industrial equipment in real-time on the Arduino-class device [2].

### D. Statistical and Threshold-Based Methods

Statistical techniques such as standard deviation thresholds, moving averages, control charts, and lightweight PCA are the most straightforward and deterministic techniques of embedded anomaly detection. They need no training, provide constant time inference, and can run on footprints of less

than 10 KB of memory. They are not as capable of finding complicated patterns as they would be with more advanced circuit-breaking techniques, but in safety-critical or ultra-low-power applications (e.g., a medical implant, an industrial alarm), they are peerless.

Recent research proves that statistical and threshold-based anomaly detection techniques are still quite useful in real-time, safety-critical, as well as low-latency industry settings. An et al. presented a constantly changing, online statistical log anomaly detection framework about the AIOps, which can be adaptable to real-time and automatically mitigates the data contamination, where improvements are up to 60% higher F1-score than the conventional static pipelines [3]. The authors in [14] compared the traditional statistical and multivariate tools for wear monitoring with force sensor streams in CNC machining and found that they have a high robustness to parameter changes and (good) predictive performance in the case of progressive failures. A comparative industrial study by Mikayilov and Gardashova [17] reported that simple yet powerful industrial lightweight unsupervised machine learning methods, such as IF and autoencoder baselines, give robust performance in noisy and high-dimensional manufacturing data, though smaller statistical models can be deployed much faster and improve operations quantitatively.

#### E. Comparative Summary

Table I summarizes the key characteristics, performance, advantages, and limitations of the four main embedded anomaly detection approaches discussed in this section, based on recent literature.

### III. DISCUSSION

The comparative landscape of embedded anomaly detection reveals a clear trade-off between model expressiveness and deployment feasibility. Tree-based methods like IF consistently emerge as the default choice for ultra-constrained microcontrollers due to their minimal memory footprint, linear-time inference, and robustness to high-dimensional sensor data. However, their inability to model temporal dynamics limits their effectiveness in sequential monitoring tasks such as vibration analysis or ECG streams. In contrast, lightweight neural networks, particularly quantized LSTM-autoencoders and compact CNNs, demonstrate superior accuracy in time-series contexts by capturing contextual dependencies, but at the cost of increased memory and computational demands that often exceed the capabilities of Cortex-M-class devices. One-class classification methods, while theoretically powerful for modeling complex decision boundaries in normal behavior, remain largely impractical for edge deployment due to their reliance on support vectors or dense kernel computations that scale poorly with data volume. Statistical and threshold-based approaches, though simplistic, retain relevance in safety-critical applications where deterministic latency and interpretability outweigh detection sophistication. Notably, the most effective real-world systems increasingly adopt hybrid strategies: using a lightweight unsupervised filter (e.g., IF) for initial anomaly

TABLE I  
COMPARATIVE ANALYSIS OF EMBEDDED ANOMALY DETECTION APPROACHES

Approach + Ref	Description	Performance	Advantages	Limitations
<b>Tree-Based Methods</b>				
Isolation Forest [11], [13], [22], [24]	Unsupervised ensemble using random partitioning; anomalies isolated in fewer splits. Federated variants enable privacy-preserving edge learning.	High accuracy/precision; low latency; microcontroller-efficient.	Low RAM, linear-time, no labels, scalable.	Struggles with temporal or clustered anomalies.
<b>One-Class Learning</b>				
OCSVM / Deep SVDD / AE [6], [15], [20], [23]	Models learn boundary around normal data (kernel, hypersphere, or reconstruction).	High recall & precision; robust to zero-day attacks.	Handles non-linear patterns; generalizes from benign-only data.	High memory (OCSVM); sensitive to thresholds; complex training.
<b>Lightweight Neural Networks</b>				
LSTM-AE, CNN, Transformer [2], [7], [12], [21]	Quantized/pruned DNNs (LSTM, CNN, Transformer) deployed via TF Lite or MicroPython.	High accuracy on time-series/multimodal data.	Captures temporal dynamics; high detection quality.	Higher RAM than trees; needs quantization & feature engineering.
<b>Statistical &amp; Threshold-Based</b>				
SPC, Moving Stats, RSM [3], [14], [17]	Rule-based: control charts, deviation thresholds, online statistical tests.	Near-zero latency; robust to parameter shifts; improves F1-score.	~10 KB RAM; deterministic; no training; safety-critical ready.	Misses complex anomalies; assumes stationarity; poor on non-Gaussian data.

screening, followed by a more resource-intensive model only when triggered. This cascaded architecture balances responsiveness with precision while conserving power, critical for battery-operated IoT nodes. Despite advances in model compression and TinyML toolchains, the gap between research prototypes and production-ready edge deployment remains wide, primarily due to inconsistent evaluation practices that prioritize accuracy over real-world constraints like latency, energy, and concept drift resilience.

### IV. RESEARCH GAPS

Current literature exhibits several critical shortcomings that hinder the practical adoption of embedded anomaly detection. First, there is a lack of standardized benchmarks that jointly evaluate detection performance alongside hardware-aware metrics such as inference latency, RAM usage, power consumption, and model update overhead. Most studies report accuracy or F1-score in isolation, often on static offline datasets, which fails to reflect the dynamic, streaming nature of real sensor data. Second, the challenge of concept drift, where the definition of “normal” evolves due to aging hardware, environmental changes, or software updates, is rarely addressed with viable on-device adaptation mechanisms. Existing methods either assume stationarity or require cloud-assisted retraining, breaking the premise of true edge auton-

omy. Third, the evaluation of adversarial robustness is almost entirely absent; lightweight models are highly susceptible to input perturbations, yet few works consider security-efficacy trade-offs. Fourth, there is minimal investigation into cross-platform portability: a model optimized for TensorFlow Lite Micro on an ARM Cortex-M4 may not translate efficiently to ESP32 or RISC-V architectures without significant re-engineering. Finally, the human-in-the-loop aspect, such as explainability for field engineers or configurable false-positive tolerance, is overlooked, reducing trust in autonomous anomaly alerts in industrial settings.

## V. FUTURE WORK

To bridge these gaps, future research should prioritize the development of holistic, hardware-aware evaluation frameworks that enforce joint optimization of accuracy, latency, memory, and energy, ideally integrated into community-driven initiatives like MLPerf Tiny. Online learning techniques must be co-designed with anomaly detectors to enable continuous adaptation to concept drift without catastrophic forgetting or excessive computational overhead; this includes exploring incremental PCA, streaming autoencoders, or federated one-class models that update only split thresholds or reconstruction baselines. Bio-inspired optimization approaches could provide new adaptation paradigms. Just as bacterial foraging optimization algorithms mimic *E. coli* behavior [16], similar mechanisms could enable anomaly detectors to dynamically adjust decision boundaries while navigating concept drift with minimal energy expenditure. Hardware-software co-design should be advanced through tighter integration of neural architecture search (NAS) with embedded constraints, yielding models that are natively quantized, sparsified, and memory-optimized during training. Hybrid optimization frameworks offer promising directions. Techniques that integrate particle swarm optimization with artificial potential fields demonstrate how continuous recalculation of optimal paths, while dynamically adjusting to environmental changes and avoiding obstacles during replanning, can reduce computation time while maintaining efficiency [8]. Such approaches could inspire energy-aware model updating strategies that dynamically adjust anomaly thresholds with minimal overhead. Additionally, research into lightweight adversarial training or input sanitization layers could enhance robustness while preserving efficiency. Temporal adaptation mechanisms represent another frontier. Architectures combining Long Short-Term Memory networks with specialized attention processors can highlight important behavioral patterns over time while filtering irrelevant data [18]. Crucially, unlike conventional systems that rely on fixed features, such models can adapt dynamically to new and evolving anomaly patterns, providing resilience against emerging threats without cloud dependency. Standardized deployment pipelines, supporting seamless conversion from PyTorch or TensorFlow to MCU-compatible formats with automatic memory budgeting, would accelerate real-world adoption. Attention mechanisms and transformer architectures represent promising directions for next-generation

embedded anomaly detection. Recent advances in sparse attention frameworks [9] demonstrate how computational efficiency can be maintained while preserving contextual awareness in sequential data. Similarly, hybrid CNN-Transformer models [10] illustrate the potential for combining local feature extraction with global relationship modeling. Future research should investigate hardware-aware implementations of these architectures that maintain their expressive power while respecting the stringent memory and latency constraints of edge devices. Finally, human-centered design principles must be incorporated: anomaly scores should be interpretable (e.g., via feature attribution on microcontrollers), and systems should support operator feedback loops to refine thresholds or label edge cases, enabling semi-supervised lifelong learning directly on the device.

## VI. CONCLUSION

This survey has looked over the terrain of real-time anomaly detection in embedded and edge IoT systems, focusing on techniques that balance the detection effectiveness and stringent hardware limitations. Approaches based on trees, specifically IFs, provide a viable option that can be deployed on ultra-constrained microcontrollers, can have low latency (less than 50 ms), low memory footprint (less than 160 KB), and achieve high performance without labeled data. Deep autoencoders and optimized OCSVM implementations, which use a single class, are more accurate and resistant to more complicated patterns, but require more resources and can be used on edge hardware such as Raspberry Pi or Jetson Nano. The lightweight neural networks (e.g., quantized LSTM-AEs) are intermediate solutions, which are capable of temporal dynamics in time-series data but still can be deployed on TinyML toolchains. Conversely, statistical and threshold-based approaches, although basic, are still useful in safety-critical or deterministic real-world environments where human readability and zero-training execution are more important than detection advanced detection capabilities.

Most importantly, there is no single set of approaches that is optimal in all situations. The best selection is based on the hardware to target, data modality, the presence of labeled anomalies, and real-time issues. The subsequent stage in co-design is the evolution of online learning with concept drift, benchmark standardization with latency and energy metrics, and explainability features to create operator trust. Lightweight, adaptive, and hardware-aware anomaly detection will continue to be a vital component of ensuring the intelligent edge as the IoT deployments become increasingly large and critical.

## REFERENCES

- [1] Deepak Adhikari, Wei Jiang, Jinyu Zhan, Danda B Rawat, and Asmita Bhattarai. Recent advances in anomaly detection in internet of things: Status, challenges, and perspectives. *Computer Science Review*, 54:100665, 2024.

- [2] Md Nur Amin, Lea Hubner, Fatih S Bayram, and Alexander Jesser. Real-time anomaly detection with lstm-autoencoder network on micro-controllers for industrial applications. In *Proceedings of the 2024 8th International Conference on Graphics and Signal Processing*, pages 42–46, 2024.
- [3] Lu An, An-Jie Tu, Xiaotong Liu, and Rama Akkiraju. Real-time statistical log anomaly detection with continuous aiops learning. In *CLOSER*, pages 223–230, 2022.
- [4] Mattia Antonini, Miguel Pincheira, Massimo Vecchio, and Fabio Antonelli. An adaptable and unsupervised tinyml anomaly detection system for extreme industrial environments. *Sensors*, 23(4):2344, 2023.
- [5] Andres J Aparcana-Tasayco, Xianjun Deng, and Jong Hyuk Park. A systematic review of anomaly detection in iot security: towards quantum machine learning approach. *EPJ Quantum Technology*, 12(1):1–39, 2025.
- [6] Aya G Ayad, Mostafa M El-Gayar, Noha A Hikal, and Nehal A Sakr. Efficient real-time anomaly detection in iot networks using one-class autoencoder and deep neural network. *Electronics*, 14(1):104, 2024.
- [7] Olufunbi Babalola, Olaitan Miriam Olufisayo Raji, Jamiu Olamilekan Akande, Abdullahi Olalekan Abdulkareem, Vincent Anyah, Adeladan Samson, and Steve Folorunso. Ai-powered cybersecurity in edge computing: Lightweight neural models for anomaly detection. 2024.
- [8] Abdelmadjid Benmachiche, Makhlof Derdour, Moustafa Sadek Kahil, Mohamed Chahine Ghanem, and Mohamed Deriche. Adaptive hybrid pso-apf algorithm for advanced path planning in next-generation autonomous robots. *Sensors*, 25(18):5742, 2025.
- [9] Soundes Oumaima Boufaïda, Abdelmadjid Benmachiche, Makhlof Derdour, Majda Maatallah, Moustafa Sadek Kahil, and Mohamed Chahine Ghanem. Tsa-gru: a novel hybrid deep learning module for learner behavior analytics in moocs. *Future Internet*, 17(8):355, 2025.
- [10] Ines Boutabia, Abdelmadjid Benmachiche, Akram Bennour, Ali Abdelatif Betouil, Makhlof Derdour, and Fahad Ghabban. Hybrid cnn-vit model for student engagement detection in open classroom environments. *SN Computer Science*, 6(6):684, 2025.
- [11] Yang Cao, Haolong Xiang, Hang Zhang, Ye Zhu, and Kai Ming Ting. Anomaly detection based on isolation mechanisms: A survey. *Machine Intelligence Research*, pages 1–17, 2025.
- [12] Jingjing Chang, Peining Zhen, Xiaotao Yan, Yixin Yang, Ziyang Gao, and Haibao Chen. Mematr: An efficient and lightweight memory-augmented transformer for video anomaly detection. *ACM Transactions on Embedded Computing Systems*, 24(3):1–26, 2025.
- [13] Wilson Chua, Arsenn Lorette Diamond Pajas, Crizelle Shane Castro, Sean Patrick Panganiban, April Joy Pasuquin, Merwin Jan Purganan, Rica Malupeng, Divine Jessa Pingad, John Paul Orolfo, Haron Hakeen Lua, et al. Web traffic anomaly detection using isolation forest. In *Informatics*, volume 11, page 83. MDPI, 2024.
- [14] Yuanzhi Huang, Eamonn Ahearne, Szymon Baron, and Andrew Parnell. An evaluation of methods for real-time anomaly detection using force measurements from the turning process. *arXiv preprint arXiv:1812.09178*, 2018.
- [15] Abdulkarim Katbi and Riadh Ksantini. One-class iot anomaly detection system using an improved interpolated deep svdd autoencoder with adversarial regularizer. *Digital Signal Processing*, 162:105153, 2025.
- [16] Amina Makhlof, Abdelmadjid Benmachiche, and Ines Boutabia. Enhanced autonomous mobile robot navigation using a hybrid bfo/pso algorithm for dynamic obstacle avoidance. *Informatica*, 48(17), 2024.
- [17] Kanan Mikayilov and Latafat Gardashova. Modern anomaly detection methods in industry: A comparative analysis of machine learning algorithms and their application to improve the efficiency of manufacturing processes. *Science*, 14:100154, 2025.
- [18] Brahim Khalil Sedraoui, Abdelmadjid Benmachiche, Akram Bennour, Amina Makhlof, Makhlof Derdour, and Fahad Ghabban. Lstm-swap: A hybrid deep learning model for cheating detection. *SN Computer Science*, 6(7):798, 2025.
- [19] Brahim Khalil Sedraoui, Abdelmadjid Benmachiche, Amina Makhlof, and Chaouki Chemam. Intrusion detection with deep learning: A literature review. In *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–8. IEEE, 2024.
- [20] Tongxin Shi, Roy A McCann, Ying Huang, Wei Wang, and Jun Kong. Malware detection for internet of things using one-class classification. *Sensors*, 24(13):4122, 2024.
- [21] Gawsalyan Sivapalan, Koushik Kumar Nundy, Soumyabrata Dev, Barry Cardiff, and Deepu John. Annet: A lightweight neural network for ecg anomaly detection in iot edge sensors. *IEEE Transactions on Biomedical Circuits and Systems*, 16(1):24–35, 2022.
- [22] Pavle Vasiljevic, Milica Matic, and Miroslav Popovic. Federated isolation forest for efficient anomaly detection on edge iot systems. *arXiv preprint arXiv:2506.05138*, 2025.
- [23] Kun Yang, Samory Kpotufe, and Nick Feamster. An efficient one-class svm for anomaly detection in the internet of things. *arXiv preprint arXiv:2104.11146*, 2021.
- [24] Amna Zahoor, Waseem Abbasi, Muhammad Zeeshan Babar, and Abeer Aljohani. Robust iot security using isolation forest and one class svm algorithms. *Scientific Reports*, 15(1):36586, 2025.

# SECURE VIDEO TRANSMISSION VIA UDP AND BLOCKCHAIN FOR SMART CITIES

Mohamed ElAmine Kheraifia<sup>1</sup>, Abdelatif Sahraoui<sup>1</sup>, Sourour Maalem<sup>2</sup>, Makhlof Derdour<sup>3</sup>

<sup>1</sup>Cheikh Larbi Tebessi University LAMIS Laboratory, Tebessa, 12000, Algeria

<sup>2</sup>LIAOA Laboratory, Higher Normal School of Constantine, Constantine 25000, Algeria

<sup>3</sup>University Of Oum el Bouaghi, LIAOA Laboratory, Oum el Bouaghi, 04000, Algeria

**Abstract**—The UDP protocol is widely used for fast data transmission in surveillance systems and real-time applications in smart cities due to its low latency. However, its lack of guarantees regarding delivery, order, error correction, and security limits its use in critical contexts where data integrity is essential, such as judicial evidence. To address these weaknesses, an innovative solution is proposed: the integration of blockchain with UDP. Each transmitted video frame is accompanied by metadata (frame number, timestamp, cryptographic hash) recorded via smart contracts on the blockchain. This combination allows for the reconstruction of the correct sequence, detection of lost or altered frames, and ensures reliable retransmission if necessary. It merges performance with security, enhancing the reliability of video streams in urban surveillance environments.

**Index Terms**—Surveillance system, UDP, Smart contract, Blockchain

## I. INTRODUCTION

The use of User Datagram Protocol (UDP) plays a key role in fast and efficient data transmission, particularly for video surveillance systems, urban sensors, and real-time communication applications. Thanks to its low latency and simplicity, UDP allows data packets to be sent without establishing a prior connection, making it particularly suitable for situations where speed trumps absolute reliability. For example, in an urban surveillance system, cameras can continuously send video streams to control centers without waiting for acknowledgment, thus ensuring an immediate response to incidents.

The use of User Datagram Protocol (UDP), while advantageous for real-time applications, has several notable drawbacks, particularly in critical contexts such as smart city video surveillance:

- 1) No delivery guarantee: UDP does not verify that sent packets actually arrive at their destination. Packet losses can go undetected, which is problematic for video surveillance or sensitive data.
- 2) No order control: Packets may arrive in a different order than they were sent, making correct video reconstruction difficult without a reordering mechanism.
- 3) No error detection: Unlike TCP, UDP does not offer an automatic correction or retransmission mechanism in the event of a transmission error.
- 4) No built-in security: UDP does not encrypt data or provide an authentication mechanism, making it vulnerable to attacks such as spoofing or interception.

- 5) Unsuitable for critical transmissions: For videos used as forensic evidence, data loss or undetected modifications can compromise the integrity of the information.

To overcome these issues, we propose an innovative technique that integrates the UDP protocol with blockchain technology to ensure the reliability, traceability, and integrity of transmitted data particularly in the context of video surveillance in smart cities. Our approach involves attaching a set of metadata (frame number, timestamp, cryptographic hash) to each video frame sent via UDP, with this information recorded on a blockchain through smart contracts. Thus, even though UDP does not guarantee packet delivery or order, the blockchain acts as an immutable and verifiable ledger that allows for reconstructing the correct sequence, detecting lost or altered frames, and requesting retransmission through a fallback protocol (such as TCP). This fusion of UDP's efficiency with blockchain's security not only maintains high performance in real-time video transmissions but also ensures the authenticity and legal compliance of the collected data crucial for urban security applications, judicial investigations, and the protection of citizens' privacy.

The content of this paper is organized as follows: Section II presents the related work. Section III integrates blockchain with UDP for reliable and traceable urban surveillance. Section IV presents the performance evaluation of the proposal. Section V concludes our work.

## II. RELATED WORK

The integration of User Datagram Protocol (UDP) with blockchain technology offers unique opportunities to improve data privacy, security, and communication efficiency. By leveraging the decentralized nature of blockchain, UDP can facilitate secure data transmission while ensuring compliance with privacy regulations. The following sections describe the key aspects of this integration.

Shawn et al. [1] focused on a computer server that facilitates data transfer between packet-based systems and blockchain-based units, emphasizing the data analysis and confirmation processes. Jong [2] presented a network protocol for blockchain that enhances network packets, potentially including UDP. This protocol aims to address the centralized weaknesses of current infrastructures, enabling trustless interoperability and encouraging participation in decentralized

applications through robust security and performance. Paper [3] focused on secure communication solutions using blockchain technology and cryptography, with an emphasis on decentralized applications and data security rather than specific transport layer protocols like UDP. Lai et al. [4] does not specifically address UDP in relation to blockchain. However, it does address broadcast protocols that are essential for efficient data dissemination in blockchain networks, including various transport layer protocols like UDP for transaction propagation. Aggarwal et al. [5] discussed the role of blockchain in improving network security and mitigating risks such as DDoS attacks, which may involve various network protocols, including UDP. Nair et al. [6] focused on enhancing packet transmission security for IoT devices using a blockchain-inspired method called PacketChain, which improves security against various cyber attacks in constrained environments. Shu et al. [7] discusses a data packet asynchronous forwarding method for blockchain networks, focusing on improving synchronization efficiency and reducing flow pressure during data packet forwarding.

### III. INTEGRATION OF BLOCKCHAIN WITH UDP FOR RELIABLE AND TRACEABLE URBAN SURVEILLANCE

Our approach involves associating each video frame sent via UDP with a set of essential metadata, such as the frame number, the precise timestamp, and a cryptographic fingerprint (generated using secure hash functions such as SHA-256). This information is automatically recorded on a blockchain through the execution of smart contracts, guaranteeing immutability and traceability of the data.

Thus, even though the UDP protocol, due to its disconnected nature, guarantees neither packet delivery nor their order, our system allows for the integrity and authenticity of frames to be verified at any time. If a missing, corrupted, or duplicated frame is detected, the system can initiate a recovery mechanism based on a complementary protocol (such as TCP or QUIC) to re-request the affected frames without interrupting the main stream.

This fusion of UDP's speed, blockchain's cryptographic reliability, and smart contract flexibility offers an ideal solution for contexts where low latency and high security are both critical. It meets the growing demands of smart cities, where real-time surveillance must coexist with personal data protection, legal evidence preservation, and resilience against cyberattacks.

### IV. EVALUATION AND RESULTS

In our experimental approach to improving the UDP protocol, we implement a system in which each video frame is transmitted in a separate packet, and its corresponding frame number, timestamp, and hash are recorded on the blockchain. This mechanism allows the receiver to identify missing frames by cross-referencing the received timestamps with those stored on the blockchain, thus facilitating targeted retransmission requests.

The experimental results in Table 1 demonstrate that optimizing the UDP protocol by recording blockchain timestamps

TABLE I  
A COMPARISON BETWEEN TRADITIONAL UDP AND  
BLOCKCHAIN-ENHANCED UDP

Method	Frames Sent	Average Packet Loss	Recovery	Recovery Delay
UDP	1000 Frames	2%	0	0
UDP+Blockchain	1000 Frames	2%	90%	0.364MS

significantly improves the reliability of video streaming. While the standard UDP protocol suffers from irrecoverable frame drops, the proposed method guarantees secure, traceable, and recoverable transmission, making it ideal for surveillance.

### V. CONCLUSION

In the future, this system can be enhanced by integrating digital watermarking techniques, lightweight encryption algorithms, and artificial intelligence models to further strengthen the protection, automatic classification, and contextual verification of videos in smart urban environments.

### REFERENCES

- [1] Shawn, S., Lindsey, W., & Michelsen, A. (2020). Architecture for facilitating data transfer for blockchain-based units in packet-based systems.
- [2] Jong, K. (2019). Network protocol for blockchain based network packets.
- [3] Blockchain and Cryptography Communication System. (2023). International Journal For Science Technology And Engineering, 11(4), 2527–2532. <https://doi.org/10.22214/ijraset.2023.50702>
- [4] Lai, Y., Liu, Y., Luo, H., Sun, G., Cheng, C., Yu, H., Atiquzzaman, M., & Dustdar, S. (2025). Broadcast Protocols in Blockchain Networks — Accelerating Block and Transaction Propagation: A Review. <https://doi.org/10.36227/techrxiv.172565468.81767780/v2>
- [5] Aggarwal, P., Thamaraimanalan, T., Logeshwaran, J., Shukla, R., Vishwakarma, P., & Aeri, M. (2024). Exploring the Synergy between Network Security and Blockchain Technology. 1–6. <https://doi.org/10.1109/icrito61523.2024.10522287>
- [6] Nair, G. (2023, May). PacketChain: a blockchain-inspired method for enhanced security of packet communication of highly constrained IoT wearable devices. In 2023 International Conference on Control, Communication and Computing (ICCC) (pp. 1–6). IEEE.
- [7] Shu, Shang. Data Packet Asynchronous Forwarding Method and System, Data Processing System and Consensus Node Terminal. 6 Sept. 2019.

# Synthesizing Brain Images with GANs: A Review of Methods and Applications in Medical Imaging

Khadija Rais<sup>✉</sup>

Laboratory of Mathematics, Informatics  
and Systems (LAMIS)

Echahid Cheikh Larbi Tebessi University  
Tébessa, 12002, Algeria  
khadija.rais@univ-tebessa.dz

Mohamed Amroune<sup>✉</sup>

Laboratory of Mathematics, Informatics  
and Systems (LAMIS)

Echahid Cheikh Larbi Tebessi University  
Tébessa, 12002, Algeria  
mohamed.amroune@univ-tebessa.dz

Mohamed Yassine Haouam<sup>✉</sup>

Laboratory of Mathematics, Informatics  
and Systems (LAMIS)

Echahid Cheikh Larbi Tebessi University  
Tébessa, 12002, Algeria  
mohamed.haouam@univ-tebessa.dz

**Abstract**—Generative Adversarial Networks (GANs) have emerged as powerful tools for synthesizing medical images, particularly for brain tumor analysis where data scarcity and class imbalance pose significant challenges. This review comprehensively examines GAN-based approaches for brain image synthesis across multiple dimensions: classification, segmentation, synthetic image generation, class integrity preservation, and explainable AI. We analyze various GAN architectures including traditional GANs, Deep Convolutional GANs (DCGANs), Wasserstein GANs (WGANs), and their variants, alongside hybrid approaches that combine GANs with other deep learning techniques. The review also addresses critical concerns regarding the preservation of diagnostic features in synthetic images and the interpretability of GAN-based medical imaging systems. By synthesizing insights from recent literature, we highlight both the remarkable progress made and the persistent challenges in applying GANs to brain tumor imaging, providing a foundation for future research directions in this rapidly evolving field.

**Index Terms**—Generative Adversarial Networks, Brain Tumor Imaging, Medical Image Synthesis, Data Augmentation, Class Integrity, Explainable AI, Image Classification, Image Segmentation

## I. INTRODUCTION

Medical imaging analysis has seen transformative advances with the advent of deep learning techniques, particularly in the domain of brain tumor detection and characterization. However, developing robust deep learning models for medical applications faces significant hurdles including limited annotated datasets, class imbalance, privacy concerns, and the high cost of data acquisition and expert annotation [1]. These challenges are especially pronounced in brain tumor imaging, where rare tumor types may be severely underrepresented in available datasets.

Generative Adversarial Networks (GANs) have emerged as a promising solution to these challenges by enabling the synthesis of realistic medical images that augment training datasets while preserving critical diagnostic features. Since their introduction by Goodfellow et al., GANs have been adapted and extended for numerous medical imaging applications, demonstrating their capacity to generate high-fidelity synthetic images that capture the complex anatomical and pathological characteristics of brain structures and tumors.

This review provides a comprehensive examination of GAN-based approaches for synthesizing brain images, with particular emphasis on their applications in classification, segmentation, and data augmentation. We further explore critical aspects such as the preservation of class integrity in generated images and the interpretability of GAN-based systems through explainable AI techniques. By synthesizing insights from both foundational and cutting-edge research, this review aims to provide researchers and practitioners with a clear understanding of the current state-of-the-art, methodological considerations, and future directions in GAN-based brain image synthesis.

The remainder of this paper is organized as follows: Section II examines GAN-based approaches for brain tumor classification; Section III reviews GAN applications in brain tumor segmentation; Section IV discusses synthetic image generation techniques; Section V addresses class integrity in synthetic images; Section VI explores explainable AI methods for GAN-based medical imaging; Section VII outlines key challenges and future directions; and Section VIII concludes the review.

## II. CLASSIFICATION APPROACHES

Brain tumor classification is a critical task in medical diagnosis, requiring precise differentiation between tumor types such as gliomas, meningiomas, and pituitary tumors, as well as distinguishing between tumor and non-tumor cases. GANs have been effectively leveraged to enhance classification performance through data augmentation and feature learning.

### A. Data Augmentation for Classification

Data augmentation using GANs has proven particularly valuable for addressing class imbalance in brain tumor datasets. Sahoo and Mishra [2] demonstrated that Progressive Growing GANs (PGGANs) outperform traditional augmentation techniques for brain tumor classification, generating synthetic images that preserve critical diagnostic features while expanding the training dataset. Their approach achieved significant improvements in classification metrics by generating high-quality images for underrepresented classes.

Nag et al. [3] proposed TumorGANet, a comprehensive framework combining transfer learning and GAN-based data

augmentation for brain tumor classification. Their approach employs ResNet50 for feature extraction and GANs for generating synthetic images, resulting in enhanced model robustness and accuracy across four classes of brain tumors. The integration of explainable AI techniques provided transparency in the classification decisions, addressing a critical need in medical applications.

Rais et al. [4] compared various generative techniques including GANs and a discriminator-enhanced Variational Autoencoder (Disc-VAE) for medical image generation. Their findings indicated that GANs generally produce higher quality images with better preservation of diagnostic features compared to VAE-based approaches, though the optimal choice depends on the specific dataset and application requirements.

### B. Hybrid Architectures for Classification

Recent research has focused on developing hybrid architectures that combine the strengths of GANs with other deep learning paradigms. Agarwal et al. [5] presented an advanced framework integrating ResNet50 with GAN-driven data augmentation specifically designed for detecting rare tumor cases. Their approach addressed challenges such as irregular tumor shapes, enhancement patterns, calcifications, and necrotic regions by generating diverse synthetic examples of these rare features.

Sun et al. [6] introduced MM-GAN, a 3D GAN architecture specifically designed for MRI data augmentation and segmentation. While primarily focused on segmentation, their approach also demonstrated significant improvements in classification tasks by generating volumetric synthetic data that preserves the 3D structural relationships critical for accurate diagnosis.

## III. SEGMENTATION APPROACHES

Brain tumor segmentation involves precisely delineating tumor boundaries and sub-regions within MRI scans, a task that is both clinically crucial and technically challenging due to the irregular shapes, heterogeneous appearances, and variable locations of tumors.

### A. GAN-Based Segmentation Frameworks

Narayanan et al. [7] proposed an automated brain tumor segmentation system using GAN augmentation and an optimized U-Net architecture. Their approach employs a two-stage process: first, a DCGAN network generates binary tumor masks that are overlaid on healthy brain images; second, a pix2pix GAN performs style transfer to create realistic synthetic images. These synthetic paired images augment the training dataset for an optimized U-Net segmentation model enhanced with residual blocks, significantly improving segmentation performance.

Sun et al. [6] developed MM-GAN, which translates label maps to 3D MR images while preserving anatomical correctness. Their approach demonstrated particular effectiveness in data-limited scenarios, improving dice scores for whole tumor and tumor core segmentation. Additionally, MM-GAN

showed promise for patient privacy protection, as models trained exclusively on synthetic data could be fine-tuned with minimal real data while maintaining high performance.

### B. Advanced Segmentation Techniques

Mukherjee et al. [8] introduced AGGrGAN, an innovative approach that aggregates outputs from multiple GAN models (two DCGAN variants and a WGAN) and applies style transfer to enhance image quality. While primarily focused on image generation, their approach demonstrated improved segmentation performance when the generated images were used for training. The aggregation of multiple GAN outputs captured diverse features that single GANs might miss, particularly important for complex brain tumor structures.

Li et al. [1] proposed TumorGAN, a multi-modal data augmentation framework specifically designed for brain tumor segmentation. Their approach leverages the complementary information present in different MRI modalities (T1, T1ce, T2, FLAIR) to generate realistic synthetic images that enhance segmentation performance across all modalities.

## IV. SYNTHETIC IMAGE GENERATION TECHNIQUES

The generation of realistic, diagnostically valuable synthetic brain images requires sophisticated architectures that preserve both global structure and local pathological details. This section reviews key techniques and architectural innovations in GAN-based medical image synthesis.

### A. GAN Architectures for Medical Imaging

Traditional GANs face challenges in generating high-resolution medical images due to training instability and mode collapse. Deep Convolutional GANs (DCGANs) address these issues through architectural improvements including convolutional layers, batch normalization, and specific activation functions [1]. DCGANs have been widely adopted for brain image synthesis due to their ability to generate higher quality images with better training stability.

Wasserstein GANs (WGANs) and their variants offer further improvements through the use of Wasserstein distance as the loss function, providing more stable training dynamics and better convergence properties [8]. The incorporation of gradient penalty (WGAN-GP) has proven particularly effective for medical image generation, as it enforces the Lipschitz constraint necessary for proper Wasserstein distance estimation.

### B. Advanced Generation Techniques

Recent research has explored hybrid approaches that combine the strengths of different generative models. Mukherjee et al. [8] developed AGGrGAN, which aggregates outputs from multiple GAN models and applies style transfer to enhance the realism of generated images. Their approach demonstrated that combining different GAN architectures could capture complementary features, resulting in higher quality synthetic images with better preservation of diagnostic characteristics.

Diffusion models have recently emerged as promising alternatives to GANs for medical image synthesis. Peng et

al. [4] demonstrated that lightweight diffusion probabilistic models (LW-DDPM) can outperform GAN-based methods for brain MRI synthesis in some contexts, offering better training stability and mode coverage while maintaining high image quality.

## V. CLASS INTEGRITY IN SYNTHETIC IMAGES

A critical concern in GAN-based medical image synthesis is whether the generated images preserve the diagnostic features of their original classes. Class integrity refers to the preservation of clinically relevant characteristics in synthetic images, ensuring they accurately represent the pathologies they are intended to model.

### A. Evaluating Class Integrity

Rais et al. [9] investigated class integrity in GAN-generated brain tumor images using a multi-faceted approach. They employed CNN classifiers to evaluate whether synthetic images were correctly classified into their original classes, Grad-CAM for visual explanation of classification decisions, and SSIM scores to quantify structural similarity between real and synthetic images. Their findings indicated that GANs can largely maintain class integrity, with CNN classifiers achieving reasonable accuracy on synthetic images and Grad-CAM highlighting anatomically relevant regions.

Traditional evaluation metrics such as Fréchet Inception Distance (FID) and Structural Similarity Index (SSIM) provide quantitative measures of image quality but do not directly assess class integrity. Recent research has emphasized the need for domain-specific evaluation frameworks that incorporate clinical expertise to validate the diagnostic utility of synthetic images.

### B. Preserving Diagnostic Features

Maintaining class integrity requires GAN architectures that explicitly model the diagnostic features of different tumor types. Attention mechanisms have proven valuable in this regard, as they enable the generator to focus on clinically relevant regions during image synthesis [8]. Style transfer techniques can further enhance the preservation of texture and morphological characteristics that are critical for diagnosis.

Semi-supervised approaches that incorporate limited labeled data during training have shown promise for improving class integrity. These methods guide the generator to produce images that not only appear realistic but also contain features relevant to specific diagnostic classes. Qi et al. [1] proposed SAG-GAN, a semi-supervised attention-guided GAN that significantly improved classification accuracy by focusing on diagnostically relevant image regions.

## VI. EXPLAINABLE AI IN GAN-BASED MEDICAL IMAGING

The clinical adoption of GAN-based medical imaging systems requires transparency and interpretability to build trust among healthcare professionals. Explainable AI (XAI) techniques provide insights into model decisions and generated images, addressing critical concerns about reliability and safety.

### A. Interpretability of Generated Images

Grad-CAM (Gradient-weighted Class Activation Mapping) has emerged as a valuable tool for interpreting GAN-generated medical images. As demonstrated by Rais et al. [9], Grad-CAM can highlight the regions of synthetic images that most strongly influence classification decisions, allowing clinicians to verify that the model focuses on anatomically and pathologically relevant features.

LIME (Local Interpretable Model-agnostic Explanations) provides another approach for explaining GAN outputs. Nag et al. [3] employed LIME to generate heatmaps that highlight the features most influential in tumor classification decisions, making the model's reasoning transparent to clinicians. These explanations help identify potential artifacts or errors in generated images that might affect diagnostic accuracy.

### B. Model Transparency and Trust

Beyond explaining individual decisions, XAI techniques can enhance the overall transparency of GAN-based systems. Attention visualization methods reveal which regions of input images most strongly influence the generator's output, providing insights into how the model processes diagnostic information.

Uncertainty quantification is another critical aspect of explainable GANs for medical imaging. Techniques that estimate the confidence of generated images can help clinicians identify potentially unreliable synthetic examples. For instance, ensemble methods that combine multiple generator outputs can provide uncertainty estimates based on the variance between different generations of the same input.

Recent research has also explored counterfactual explanations for GANs, showing how minimal changes to input conditions would alter the generated output. These explanations help clinicians understand the relationship between pathological features and their visual manifestations in generated images.

## VII. CHALLENGES AND FUTURE DIRECTIONS

Despite significant advances, several challenges remain in applying GANs to brain image synthesis:

**Quality and Diversity:** Balancing image quality with diversity remains challenging. High-quality synthetic images sometimes lack the diversity needed for robust model training, while diverse generations may sacrifice diagnostic accuracy. Future work should focus on architectures that simultaneously optimize both quality and diversity.

**3D Synthesis:** Most current approaches generate 2D slices rather than full 3D volumes. Developing efficient methods for 3D brain image synthesis that preserve spatial coherence across slices is crucial for clinical applications.

**Multi-modal Generation:** Brain tumor diagnosis often requires analysis of multiple MRI modalities (T1, T1ce, T2, FLAIR). Generating consistent synthetic images across these modalities while preserving their complementary diagnostic information presents significant technical challenges.

**Clinical Validation:** Rigorous clinical validation of synthetic images by domain experts remains essential but under-explored. Future work should establish standardized evaluation protocols involving radiologists and neuro-oncologists.

**Privacy and Ethics:** As synthetic data becomes more realistic, concerns about potential misuse and privacy implications increase. Research on differentially private GANs and other privacy-preserving techniques is needed to address these concerns.

Promising future directions include the development of foundation models pre-trained on diverse medical imaging datasets, integration of domain knowledge into GAN architectures through physics-informed generators, and federated learning approaches that enable collaborative model training across institutions while preserving data privacy.

## VIII. CONCLUSION

GANs have demonstrated remarkable potential for synthesizing brain images that address critical challenges in medical imaging, including data scarcity, class imbalance, and privacy concerns. This review has comprehensively examined GAN-based approaches across multiple dimensions: classification, segmentation, synthetic image generation, class integrity preservation, and explainable AI.

The literature reveals significant progress in GAN architectures specifically designed for medical imaging, with innovations such as attention mechanisms, style transfer, and hybrid approaches substantially improving the quality and diagnostic utility of synthetic images. However, preserving class integrity remains a critical concern that requires ongoing attention through specialized evaluation frameworks and architectural innovations.

Explainable AI techniques have emerged as essential components of GAN-based medical imaging systems, providing transparency that builds clinician trust and enhances clinical utility. The integration of methods like Grad-CAM and LIME into GAN workflows represents an important step toward clinically viable synthetic image generation.

Future research should focus on addressing persistent challenges in 3D synthesis, multi-modal generation, and rigorous clinical validation. The development of standardized evaluation protocols involving domain experts will be crucial for translating GAN-based image synthesis from research to clinical practice. As these challenges are addressed, GANs are poised to play an increasingly important role in advancing brain tumor diagnosis, treatment planning, and clinical research through the generation of high-quality synthetic medical images.

## ACKNOWLEDGMENT

This work was supported by the Laboratory of Mathematics, Informatics and Systems (LAMIS) at Echahid Cheikh Larbi Tebessi University, Tebessa, Algeria.

## REFERENCES

- [1] K. Rais, M. Amroune, M. Y. Haouam, and I. Bendib, "Comparative study of data augmentation approaches for improving medical image classification," in *2023 International Conference on Computational Science and Computational Intelligence (CSCI)*. IEEE, 2023, pp. 1226–1234.
- [2] S. Sahoo and S. Mishra, "A comparative analysis of pggan with other data augmentation technique for brain tumor classification," in *2022 International Conference on Advancements in Smart, Secure and Intelligent Computing (ASSIC)*, 2022, pp. 1–7.
- [3] A. Nag, H. Mondal, M. M. Hassan, T. Al-Shehari, M. Kadrie, M. Al-Razgan, T. Alfakih, S. Biswas, and A. K. Bairagi, "Tumorganet: A transfer learning and generative adversarial network-based data augmentation model for brain tumor classification," *IEEE Access*, vol. 12, pp. 103 060–103 081, 2024.
- [4] K. Rais, M. Amroune, and M. Y. Haouam, "Medical image generation techniques for data augmentation: Disc-vae versus gan," in *2024 6th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*. IEEE, 2024, pp. 1–8.
- [5] M. Agarwal, A. Abhisikta, P. K. Mallick, A. Kumar Jagadev, and B. Sahoo, "Advanced deep learning framework for mri brain tumor detection: Resnet50 and gan-driven data augmentation for rare tumor analysis," in *2025 International Conference on Emerging Systems and Intelligent Computing (ESIC)*, 2025, pp. 853–858.
- [6] Y. Sun, P. Yuan, and Y. Sun, "Mm-gan: 3d mri data augmentation for medical image segmentation via generative adversarial networks," in *2020 IEEE International Conference on Knowledge Graph (ICKG)*, 2020, pp. 227–234.
- [7] S. J. Narayanan, A. S. Anil, C. Ashtikar, S. Chunduri, and S. Saman, "Automated brain tumor segmentation using gan augmentation and optimized u-net," in *Frontiers of ICT in Healthcare: Proceedings of EAIT 2022*. Springer, 2023, pp. 635–646.
- [8] D. Mukherjee, P. Saha, D. Kaplun, A. Sinitca, and R. Sarkar, "Brain tumor image generation using an aggregation of gan models with style transfer," *Scientific Reports*, vol. 12, no. 1, p. 9141, 2022.
- [9] K. Rais, M. Amroune, M. Y. Haouam, and A. Benmachiche, "Evaluating class integrity in gan-generated synthetic medical datasets," in *7th International Conference on Networking and Advanced Systems (ICNAS'2025)*, University Chadli Bendjedji, El Tarf, Algeria, 10 2025.

# Towards a System for Detection, Prevention and Resilience against Data Injection Attacks on Autonomous Drones

Moustafa Sadek Kahil<sup>1</sup>, Hamda Slimi<sup>2</sup>, Sourour Maalem<sup>3</sup>, Amira Bouamrane<sup>1</sup>, Makhoul Derdour<sup>1</sup>

<sup>1</sup>Artificial Intelligence and Autonomous Things Laboratory, Larbi Ben M'hidi University, Oum El Bouaghi, 04000, Algeria

<sup>2</sup>Laboratory of Mathematics, Informatics and Systems; Echahid Cheikh Larbi Tebessi University, Tebessa, Algeria

<sup>3</sup>LIAOA Laboratory, Higher Normal School of Constantine, Constantine 25000, Algeria

**Abstract**—Embedded systems, such as those used in autonomous drones, often rely on complex hardware and software architectures. These architectures integrate semiconductors to manage sensors, communications, and real-time computing. However, the security of these systems is becoming critical, particularly in the face of data injection attacks, which exploit vulnerabilities in input/output interfaces, communication protocols, or data processing mechanisms. Autonomous drones, particularly, are vulnerable to these attacks because they rely on data streams from sensors (GPS, cameras, LiDAR, etc.) and wireless communications (Wi-Fi, 5G, or others). Such a targeted attack could: Disrupt a drone's navigation or mission, cause collisions or hijackings and compromise the confidentiality of collected data. This paper explores the vulnerabilities of semiconductor architectures to data injection attacks in drone scenarios, while proposing innovative solutions to detect, prevent, and make these systems resilient.

**Index Terms**—Semiconductor, IoV, IoT, cybersecurity, vulnerability, attack

## I. INTRODUCTION

Embedded systems, such as those used in autonomous drones, often rely on complex hardware and software architectures. These architectures integrate semiconductors to manage sensors, communications, and real-time computing. However, the security of these systems is becoming critical, particularly in the face of data injection attacks, which exploit vulnerabilities in input/output interfaces, communication protocols, or data processing mechanisms.

This paper addresses a critical area at the intersection of cybersecurity, embedded systems, and autonomous drones. It addresses current challenges by proposing innovative solutions for securing complex semiconductor architectures while ensuring drone reliability in sensitive scenarios.

Until recently, semiconductor security was perceived as a theoretical concern rather than a tangible threat. Governments, for example, expressed concerns about the possibility of adversaries gaining control of secure systems through hardware backdoors, whether via third-party IP or unidentified actors in the global supply chain. However, the rest of the chip industry generally paid little attention to this risk, except secure boot and firmware authentication capabilities.

However, as advanced electronics are deployed in diverse domains such as vehicles, robots, drones, medical devices, and server applications, robust hardware security is becoming

imperative. This security cannot be considered a mere “nice-to-have” feature, as security flaws in integrated circuits can compromise security, damage critical data, and disrupt business operations until the damage is assessed and the threat is remediated.

This paper explores semiconductor architectures' vulnerabilities to data injection attacks in drone scenarios while proposing innovative solutions to detect, prevent, and make these systems resilient.

Semiconductors play a key role in drone onboard architectures, but their interfaces and communication channels can be exploited for injection attacks. These attacks rely on:

- Tampering with sensor data (e.g., GPS data, altimeters, or video streams).
- Disrupting communication protocols or electronic circuits.
- Injecting malicious commands into control systems.

The main challenge lies in detecting such attacks in real time while maintaining the performance and low power consumption of the embedded systems. How can we design security mechanisms that are effective, lightweight, and capable of being integrated into complex semiconductor architectures for drones?

## II. OBJECTIVES

The objective of our system is:

- **Vulnerability Analysis:** Identify data injection attack vectors in drone semiconductor architectures and study vulnerabilities in communication channels and sensor-system interfaces.
- **Attack Model Creation:** Simulate data injection attack scenarios on autonomous drones (e.g., GPS, altimeter, or LiDAR data spoofing) and assess the impacts on navigation, autonomous decision-making, and drone control systems.
- **Detection Mechanism Design:** Develop algorithms based on real-time data flow analysis to detect abnormal behavior (e.g., machine learning to detect anomalies) and leverage semiconductor hardware capabilities to integrate security mechanisms into the hardware (e.g., Trusted Execution Environment, FPGA).

- Development of prevention and resilience mechanisms: Design countermeasures to block or mitigate detected attacks, such as secure communication protocols or redundancy mechanisms, and integrate automatic recovery solutions after an attack to ensure drone mission continuity.
- Experimental validation: Test the proposed mechanisms in simulated and real-world environments, using autonomous drone platforms and commonly used semiconductors (e.g., ARM, FPGA), and evaluate performance in terms of efficiency, computational cost, and mission impact.

### III. EXPECTED SCIENTIFIC CONTRIBUTIONS

- An in-depth analysis of semiconductor vulnerabilities to injection attacks in drone embedded systems.
- Innovative mechanisms for detecting and preventing data injection attacks.
- A contribution to the security of autonomous drones in critical scenarios, such as delivery, surveillance, or search and rescue.
- A methodology that can be generalized to other embedded systems incorporating semiconductors.

### IV. POTENTIAL APPLICATIONS

#### A. Drone security in critical sectors:

- Military or civilian surveillance.
- Autonomous delivery.
- Search and rescue missions.

#### B. Semiconductor industries:

- Development of attack-resistant semiconductors for embedded systems.

#### C. Related fields:

- Securing other embedded systems, such as autonomous cars, industrial robots, or IoT devices.

### V. CONCLUSION

This paper addresses a critical area at the intersection of cybersecurity, embedded systems, and autonomous drones. It addresses current challenges by proposing innovative solutions for securing complex architectures incorporating semiconductors, while ensuring drone reliability in sensitive scenarios.

### REFERENCES

- [1] S. Rajasekar et al., "Data injection attacks in cyber-physical systems: Vulnerabilities and countermeasures," *Computer Networks*, vol. 198, p. 108421, 2021. doi: 10.1016/j.comnet.2021.108421.
- [2] Y. Chen et al., "Adversarial and data injection attacks in autonomous systems," *Journal of Cyber-Physical Systems*, 2022.
- [3] J. Sun et al., "Real-time anomaly detection for data injection attacks in UAV systems using machine learning," 2023.