Multi-Objective Evaluation of Classification Algorithms for Internet of Things (IoT) Systems and Edge Computing

Mouataz Idrissi Khaldi [1,*], Allae Erraissi [2], Mustapha Hain [1], and Mouad Banane [1]

[1] Laboratory of Artificial Intelligence, Modeling and Computational Engineering (AIMCE), ENSAM, Hassan II University, Casablanca, Morocco
[2] Department of Information Technology, Polydisciplinary Faculty, Chouaib Doukkali University, El Jadida, Morocco
Email: mo3tazik@gmail.com (M.I.K.); erraissi.a@ucd.ac.ma (A.E.); mustapha.hain@ensam-casa.ma (M.H.); mouadbanane@gmail.com (M.B.)
*Corresponding author

***Abstract*—The rapid rise of connected systems and the Internet of Things (IoT) has led to the widespread deployment of machine learning techniques in distributed and resource-constrained environments. In this context, the selection of classification algorithms can no longer rely solely on predictive performance, but must also consider computational cost, latency, robustness, and deployment constraints. This paper proposes a multi-objective evaluation framework for selecting supervised classification algorithms in IoT-oriented connected systems. A comparative theoretical study of major classification algorithms is first conducted using a trade-off-based perspective. The framework is then validated experimentally on two complementary datasets: the Human Activity Recognition (HAR) dataset, representing sensor-based IoT applications, and the University of New South Wales Network-Based 15 (UNSW-NB15) dataset, representing network-based IoT environments. The evaluation incorporates both predictive and computational metrics, including inference time, in order to better reflect real deployment constraints. The results confirm that no single algorithm dominates all criteria and that model selection strongly depends on the target deployment scenario, whether in constrained IoT nodes, edge computing platforms, or cloud environments. The proposed framework provides a structured decision-support tool for selecting classification algorithms in connected systems, highlighting the importance of multi-objective trade-offs for real-world IoT deployments.**

***Keywords*—Internet of Things (IoT), connected systems, supervised classification, multi-objective evaluation, algorithm selection, edge computing, machine learning deployment**

I. Introduction

The rapid evolution of connected systems and the Internet of Things (IoT) has profoundly transformed the way data is collected, transmitted, processed, and exploited across modern digital infrastructures. Smart sensor networks, distributed monitoring platforms, wearable devices and connected industrial environments now generate massive volumes of heterogeneous and distributed data, requiring reliable and efficient intelligent analysis mechanisms [1, 2]. In parallel, the emergence of edge and fog computing paradigms has shifted data processing closer to end devices in order to reduce latency, bandwidth consumption, and energy overhead in real-time IoT applications [3, 4].

In this context, supervised classification algorithms play a central role in enabling intelligent decision-making within connected IoT ecosystems. They are widely used for event detection, behavior identification, network monitoring, anomaly detection, predictive maintenance and decision support in distributed architectures and smart environments [5–7]. Classic models such as Naive Bayes (NB), Decision Tree (DT), Random Forest (RF), k-Nearest Neighbors (k-NN), and Support Vector Machine (SVM) remain widely adopted due to their robustness, interpretability and ability to process heterogeneous data collected from distributed sensors and network infrastructures [4, 8, 9].

However, the choice of a classification algorithm can no longer be guided solely by predictive performance. Deployment constraints such as computational cost, inference latency, energy consumption, memory footprint, scalability, and interpretability are becoming decisive factors, particularly in IoT, edge, and fog computing environments [2, 5, 10]. Recent studies on Edge Artificial Intelligence (Edge AI) and Tiny Machine Learning (TinyML) further emphasize the need for lightweight and resource-efficient machine learning models capable of operating under strict hardware and energy limitations [11]. The work presented in the proceedings of the 3rd International Conference on Connected Objects and Artificial Intelligence (COCIA’2025) clearly illustrates these challenges through various applications such as IoT network optimization, smart agriculture data

Manuscript received January 30, 2026; revised April 22, 2026; accepted July 6, 2026; published September 18, 2026.

classification, connected infrastructure security, and performance evaluation in distributed architectures [12].

In the conference version of this work, published in Springer's Lecture Notes in Networks and Systems series, an initial multi-criteria comparison framework for the main supervised classification algorithms was proposed, highlighting the trade-offs between predictive accuracy, robustness, computational complexity, and interpretability [13]. Similar trade-off-oriented analyses have recently gained increasing attention in the context of edge intelligence and resource-aware machine learning, where deployment feasibility is considered alongside predictive performance [4, 14]. However, the initial conference contribution was based mainly on theoretical and bibliographic analysis, without explicit experimental validation or detailed consideration of the operational constraints specific to connected and distributed IoT systems. Recent studies emphasize that practical deployment factors such as latency, resource allocation, and energy efficiency must be integrated into the evaluation process of machine learning models deployed at the edge [15, 16].

Despite numerous comparative studies on classification algorithms, few works simultaneously consider predictive performance, computational requirements, and deployment constraints within a unified multi-objective framework specifically designed for IoT and edge computing environments. This expanded version builds directly on this initial work while making substantial contributions. It introduces an explicit focus on the context of IoT systems and distributed architectures, a multi-objective formalization of the problem of selecting classification algorithms, and experimental validation on multiple datasets representative of IoT applications. The objective is to go beyond traditional single-criterion approaches and propose an evaluation framework capable of reflecting the real trade-offs encountered when deploying artificial intelligence solutions in constrained environments.

In particular, it integrates IoT and edge computing constraints more explicitly, proposes a formal multi-objective formulation of the algorithm selection problem, and extends the experimental validation by incorporating multiple datasets, including both sensor-based and network-based IoT scenarios. In addition, new evaluation metrics such as inference time are introduced to better reflect deployment constraints. These extensions enable a more comprehensive comparative analysis and improve the practical relevance and generalizability of the proposed framework. These contributions clearly distinguish the present work from the initial conference version and strengthen its scientific and practical contributions.

Thus, this article aims to provide a more comprehensive and realistic decision support framework for choosing supervised classification algorithms in connected systems, simultaneously taking into account predictive performance, computational requirements, and deployment constraints specific to modern IoT architectures.

Building upon the conference version, this study introduces a formal multi-objective formulation of the selection problem, along with an enhanced theoretical and experimental analysis.

II. Extended Related Work in the Context of IoT and Connected Systems

A. *Classification and Artificial Intelligence in Connected Systems*

Modern connected systems rely on the close integration between sensors, communication infrastructures, distributed computing paradigms and artificial intelligence-based analysis mechanisms. In such architectures, supervised classification plays a central role at several functional levels, including network traffic analysis, interpretation of signals from heterogeneous sensors, activity recognition, anomaly detection, predictive maintenance and filtering of critical events [1, 4]. Recent studies further demonstrate that machine learning models deployed in IoT and edge environments must simultaneously ensure predictive efficiency, scalability, low latency, and adaptability to heterogeneous and dynamic operating conditions [2, 9]. In parallel, the growing complexity of IoT ecosystems has increased the need for intelligent classification frameworks capable of supporting real-time decision-making in resource-constrained distributed environments [16].

Numerous recent contributions presented in the conference proceedings illustrate the growing integration of classification techniques into complete IoT processing pipelines, ranging from data acquisition and sensor communication to intelligent analysis and automated decision-making [12]. These studies cover a wide range of application domains, including network monitoring, intrusion detection, smart agriculture, connected infrastructure security, intelligent healthcare systems, and performance optimization in distributed and edge-based environments [12]. Similar trends are also highlighted in recent surveys on Edge AI and IoT analytics, which emphasize the increasing interdependence between machine learning performance and deployment constraints in distributed architectures [2, 4]. These works confirm that the performance of a classification model cannot be evaluated independently from its operational deployment context, particularly in resource-constrained IoT and edge computing infrastructures [11, 14].

This observation is consistent with the conclusions of our previous work [13], in which a multi-criteria comparison of the main supervised classification algorithms highlighted the absence of a universally optimal model and emphasized the necessity of simultaneously considering several evaluation dimensions such as predictive accuracy, computational complexity, robustness, interpretability, and deployment feasibility. Similar conclusions have been reported in recent studies on Edge AI and intelligent IoT systems, where the selection of machine learning models increasingly depends on the trade-offs between predictive performance, resource consumption, scalability, and inference latency [17, 18]. Recent research on lightweight and edge-deployable machine learning models further confirms that no single classifier consistently outperforms others across all IoT scenarios and operational constraints [19, 20].

B. Network Constraints, Quality of Service, and IoT Architectures

In distributed IoT environments, network traffic and Quality of Service (QoS) management is a major challenge. The large number of connected devices, heterogeneous communication protocols, dynamic traffic patterns, and latency-sensitive services make resource allocation and traffic control particularly complex. Several recent studies have examined the contribution of Software-Defined Networking (SDN), Software-Defined Internet of Things (SD-IoT) and edge-based architectures to improving dynamic traffic management, QoS-aware routing, load balancing and the overall performance of connected device networks [5, 21–24].

The contributions presented in the Lecture Notes in Networks and Systems (LNNS) proceedings of COCIA'2025 highlight the importance of parameters such as latency, resource consumption, and flexibility of control mechanisms in evaluating intelligent solutions deployed on networks [12]. These observations are consistent with recent studies on IoT-edge architectures, where latency, scalability, limited edge resources, and communication overhead are identified as key constraints for deploying intelligent services [4, 25]. These network constraints have a direct impact on the choice of machine learning algorithms, with some models being better suited than others to highly constrained contexts, particularly in edge and fog computing architectures. Recent works on real-time edge inference and TinyML further confirm that model selection must balance predictive performance with inference latency, energy consumption, memory footprint, and deployment feasibility [11, 14].

C. Artificial Intelligence and Classification in Vertical IoT Applications

The integration of artificial intelligence and supervised classification into vertical IoT applications is also widely documented. In smart agriculture, for example, several studies use image classification combined with IoT sensors to detect crop diseases, analyze the condition of plantations, and optimize yields [12, 26–28]. These systems are generally based on hybrid architectures combining edge computing and cloud processing, illustrating the need for models capable of adapting to heterogeneous infrastructures [29, 30].

Similarly, in the field of cybersecurity for connected systems, numerous studies use supervised classification techniques for intrusion detection, anomaly identification and protection of IoT communication infrastructures. These intelligent security frameworks increasingly integrate machine learning models into distributed IoT and edge environments in order to detect abnormal behaviors in real time and improve network resilience against evolving cyber threats [31–33]. Recent studies further emphasize the joint importance of predictive accuracy, robustness to noisy and heterogeneous data, scalability, low inference latency, and model generalization capabilities in highly dynamic and distributed IoT ecosystems [34, 35]. In addition, lightweight and resource-aware intrusion detection models are increasingly required to ensure efficient deployment in constrained edge and fog computing architectures [30, 36].

D. Limitations of Traditional Algorithm Comparisons

Despite the abundance of comparative studies on classification algorithms, most research still focuses on a limited number of metrics, mainly accuracy or error rate. Recent comparative studies have shown that relying on a single performance metric may lead to biased conclusions regarding model effectiveness, and that multi-metric evaluation frameworks provide a more comprehensive understanding of classifier behavior across different operational contexts [37]. Criteria related to computational complexity, scalability, interpretability, energy consumption, inference latency or deployment cost are often treated as secondary or even ignored [38]. Recent studies in edge computing and resource-constrained machine learning environments have shown that predictive performance alone is insufficient for selecting appropriate models for real-world deployment [4, 14].

This limitation has already been highlighted in our contribution presented at COCIA'2025, where a multi-criteria analysis highlighted the need to evaluate classification algorithms using several dimensions simultaneously, particularly in connected environments where technical constraints are as important as predictive performance [13]. Similar conclusions have been reported in recent comparative studies emphasizing decision-support frameworks and multi-metric evaluation strategies for machine learning model selection [39]. Recent multi-metric comparative studies further advocate decision-support frameworks capable of integrating several evaluation dimensions rather than relying solely on predictive accuracy [37]. There is therefore a clear need for structured and generic evaluation frameworks capable of guiding the choice of machine learning models in real-world, constrained IoT systems. Unlike most existing studies, which evaluate classifiers either from a predictive perspective or from a deployment perspective, the proposed framework combines both dimensions within a unified multi-objective evaluation process specifically designed for IoT and edge computing environments.

III. Multi-objective Evaluation Framework for Algorithm Selection

A. General Principle of the Proposed Framework

The proposed evaluation framework aims to structure the comparison of supervised classification algorithms around several complementary axes, including predictive performance, robustness, computational complexity, scalability, and interpretability. This multidimensional approach goes beyond traditional single-criterion evaluations, which are mainly focused on accuracy, and better reflects the real requirements of connected and distributed systems [38].

An initial multi-criteria analysis of the main classification algorithms highlighted the importance of considering these dimensions simultaneously to avoid suboptimal choices under real-world conditions in our earlier work [13]. This extended version generalizes the framework by explicitly framing it from a multi-objective-optimization perspective tailored for IoT, edge, and cloud environments.

B. *Multi-objective Formulation of the Problem*

The problem of selecting a classification algorithm can be formalized as a Multi-Objective Optimization (MOO) problem, in which several partially conflicting criteria must be optimized simultaneously [40]. More specifically, the objectives considered can be expressed as follows:

- maximize prediction quality, evaluated using metrics such as accuracy or F1-Score.
- maximize robustness to noise and data perturbations.
- minimize computational cost, including algorithmic complexity and resource consumption.
- minimize training and inference time, which is particularly critical in real-time environments.

These objectives can be summarized as maximizing accuracy and robustness, while minimizing complexity, latency, and energy consumption. Due to the conflicting nature among these criteria, a single optimal solution generally does not exist. Consequently, it is essential to investigate the trade-offs among different objectives instead of emphasizing only one criterion, which is a common limitation in conventional performance comparisons.

C. *Trade-off Analysis and Pareto Front*

In a multi-objective context, the concept of the Pareto front is a key tool for analyzing trade-offs between criteria [40]. An algorithm is said to be Pareto-optimal if there is no other algorithm capable of improving one objective without deteriorating at least one other.

Applied to the selection of classification algorithms, this approach makes it possible to identify models offering different acceptable trade-offs. Some algorithms are distinguished by high accuracy and robustness, at the cost of significant computational overhead, while others favor simplicity, speed, and low resource consumption, at the expense of some predictive performance.

This compromise-based interpretation, already initiated in our previous comparative analysis [13], is particularly well suited to IoT systems, where the choice of model depends heavily on the execution context and deployment constraints.

D. *Decision Scenarios Based on Deployment Architecture*

In order to make this framework operational, three typical decision scenarios can be distinguished based on the deployment architecture of connected systems [5]:

- Highly constrained nodes (sensors, embedded devices): priority given to simplicity, low computational cost, and low energy consumption.
- Edge computing: seeking a compromise between predictive performance, latency, and resource consumption.
- Cloud computing: priority given to accuracy, robustness, and the ability to process large volumes of data.

This typology complements theoretical analysis by providing a decision-making framework that can be directly applied to the deployment of classification algorithms in real IoT environments, as confirmed by several recent contributions to the LNNS volume [12].

E. *Operational Multi-objective Decision Strategy*

While the multi-objective formulation provides a conceptual framework, it is important to operationalize the decision process. To support practical decision-making, a weighted scoring approach can be used [40]. Each algorithm is evaluated across multiple criteria, and a global score is computed as a weighted combination of normalized metrics.

This approach allows decision-makers to adapt model selection according to deployment constraints by assigning different importance levels to criteria such as predictive performance, latency, and computational cost.

The overall score for algorithm a_i is defined as:

$$\text{Score}\,(a_i) = \sum_{j=1}^{k} w_j C_j(a_i)\,,\ w_j \geq 0\,,\ \sum_{j=1}^{k} w_j = 1$$

where k denotes the number of evaluation criteria, w_j represents the weight assigned to criterion j, and $C_j(a_i)$ denotes the normalized performance of algorithm a_i with respect to criterion j.

For instance, in edge computing scenarios, higher importance may be assigned to latency and computational efficiency, leading to the selection of lightweight models such as Decision Tree. In contrast, in cloud environments, predictive performance may be prioritized, favoring models such as Random Forest.

This simplified approach provides an interpretable alternative to more complex optimization methods while remaining suitable for practical deployment scenarios [40].

F. *Pareto-based Interpretation of Trade-offs*

In addition to weighted scoring, a Pareto-based interpretation is used to analyze trade-offs between conflicting objectives. Based on this perspective, some algorithms provide higher predictive performance at the cost of increased computational complexity, while others offer lower computational requirements with reduced accuracy.

This interpretation enables the identification of efficient trade-offs without imposing a strict ranking, which is particularly relevant in IoT environments where deployment constraints vary. This interpretation complements the experimental analysis presented in Section V.

IV. Theoretical Comparative Analysis

This section presents a theoretical comparative analysis of the main supervised classification algorithms, based on the synthetic results introduced in our previous work [13]. Rather than simply reproducing the results, the objective here is to reinterpret and enrich this analysis in light of the multi-objective optimization framework presented in Section III, while explicitly anchoring it in the context of IoT systems and connected architectures.

A. *Multi-criteria Comparative Analysis of Classification Algorithms*

Table I provides a theoretical multi-criteria comparison of five widely used classification algorithms: Naïve Bayes (NB), Decision Tree (DT), Random Forest (RF), k-Nearest Neighbors (k-NN), and Support Vector Machine (SVM). The evaluated criteria cover predictive performance, robustness to noise, computational complexity, scalability, and interpretability, following both our previously proposed framework and state-of-the-art literature practices [13].

TABLE I. Multi-criteria Comparative Analysis of Classification Algorithms Adapted From [13]

Criteria	NB	DT	RF	k-NN	SVM	Comment
Accuracy	3/5	3/5	5/5	3/5	5/5	RF and SVM are the most accurate but more expensive.
Robustness	2/5	3/5	5/5	2/5	4/5	RF is the most stable, SVM depends on hyperparameter settings.
Complexity	5/5	4/5	3/5	2/5	2/5	NB and DT are the fastest to train.
Scalability	5/5	3/5	3/5	2/5	2/5	NB is optimal for very large data sets.
Interpretability	4/5	5/5	3/5	2/5	2/5	DT is the easiest to explain.

The scores reported in Table I are based on a consolidated theoretical analysis of well-established and recent studies on classification algorithms, including foundational contributions and subsequent benchmarking efforts [13].

The theoretical scoring presented in Table I is based on a structured synthesis of existing literature on supervised classification algorithms. The scores (ranging from 1 to 5) reflect the relative performance of each algorithm with respect to specific criteria such as accuracy, interpretability, robustness, computational cost, and scalability.

These scores were not obtained through direct numerical computation, but rather through a qualitative aggregation of findings reported in prior studies, including well-established references and recent comparative works. For each criterion, the score assigned to a given algorithm represents its relative positioning compared to other models, based on consensus trends observed in the literature.

While this approach introduces a degree of subjectivity, it provides a practical and interpretable way to summarize multi-dimensional algorithm characteristics. The objective is not to provide absolute measurements, but to support comparative analysis and decision-making within a multi-objective framework.

The scoring is supported by multiple studies in machine learning and IoT contexts, including those that provide detailed evaluations of classification algorithms across different criteria [1, 5, 8, 13].

It is important to note that the scoring scale is used for comparative purposes rather than absolute evaluation, and should be interpreted in conjunction with experimental results.

The analysis in Table I highlights clear trends. RF and SVM algorithms stand out with high scores for accuracy and robustness, confirming their effectiveness for complex and noisy classification tasks. However, this performance comes with greater computational complexity and reduced interpretability, which can limit their deployment in highly constrained IoT environments.

Conversely, Naïve Bayes and Decision Tree score highly in terms of simplicity, speed of execution, and interpretability. These characteristics make them particularly suitable for scenarios where resources are limited or where decision transparency is required, although their predictive performance remains more moderate. The k-NN model appears to be an intermediate solution, penalized by computational costs and low scalability as the volume of data increases.

This comparison confirms, as already observed by Khaldi *et al.* [13], that no single algorithm dominates all of the criteria considered, which fully justifies the adoption of a multi-objective approach.

B. *Radar-based Visualization of Multi-Criteria Trade-offs*

To supplement the tabular analysis, Fig. 1 provides a radar chart visualization of the relative performance of the algorithms evaluated according to the various criteria. This graphical representation provides a concise and intuitive overview of the trade-offs inherent in each model.

Fig. 1 shows distinct profiles. RF and SVM are top performers in terms of accuracy and robustness, but score lower in complexity and interpretability. In contrast, Naïve Bayes has a profile strongly oriented towards computational efficiency and scalability, making it a relevant candidate for deployment on constrained IoT nodes. Decision Tree stands out for its good balance between interpretability and complexity, while k-NN shows notable limitations in terms of scalability.

This visualization facilitates the rapid identification of trade-offs and complements the theoretical analysis presented in Table I .

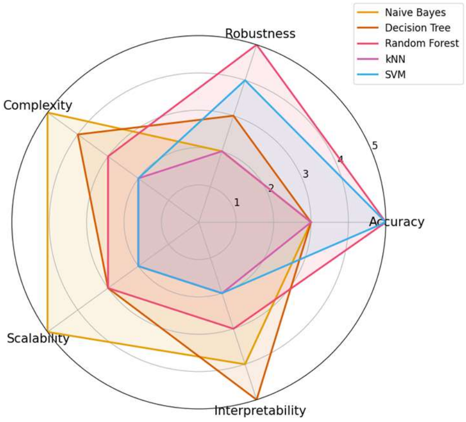

Fig. 1. Radar chart illustrating multi-criteria trade-offs between classification algorithms, adapted from Ref. [13].

C. Interpretation from a Multi-objective IoT Perspective

Placed in a multi-objective IoT-oriented perspective, the results in Table I and Fig. 1 allow us to formulate several general recommendations. Lightweight and interpretable algorithms, such as Naïve Bayes and Decision Tree, appear to be suitable solutions for embedded nodes and resource-constrained devices. Conversely, RF and SVM are more appropriate for edge or cloud environments, where computing capabilities allow for a focus on accuracy and robustness.

This interpretation is consistent with the Pareto front analysis introduced in Section III, where each algorithm can be seen as representing a specific trade-off between predictive performance and deployment constraints. Thus, the choice of an optimal algorithm depends less on an overall ranking than on the application and architectural context under consideration.

D. Discussion and Limitations

Although this comparative analysis provides a structured overview of the relative strengths and weaknesses of the algorithms studied, it is based on a theoretical and bibliographic synthesis. The scores assigned may vary depending on the nature of the data, hyperparameter settings, or execution conditions, as highlighted in several previous studies [13].

These limitations justify the need for additional experimental validation. The following section therefore introduces an empirical evaluation based on data representative of IoT applications, in order to compare the theoretical observations from Table I and Fig. 1 with concrete experimental results.

V. Experimental Validation on Multiple IoT-oriented Datasets

A. HAR Dataset Description

To complement the theoretical analysis and to assess the relevance of the proposed multi-objective evaluation framework in a realistic connected-systems context, an experimental validation was conducted using the Human Activity Recognition (HAR) Using Smartphones dataset from the UCI Machine Learning Repository [41].

This dataset is widely used in Internet of Things (IoT), wearable computing, and intelligent monitoring applications, particularly in the context of smart healthcare and human activity monitoring. Data are collected from inertial sensors, namely accelerometers and gyroscopes, embedded in smartphones or wearable devices, making this dataset a representative benchmark for evaluating classification algorithms under IoT-related constraints.

The dataset consists of 10,299 instances described by 561 extracted features, derived from time-domain and frequency-domain sensor signals. It covers six human activity classes, and is partitioned into 7209 training samples and 3090 test samples, with a stratified class distribution ensuring balanced representation across activities.

These characteristics make the HAR dataset well suited for analyzing trade-offs between predictive performance and computational cost in resource-constrained IoT environments.

B. Additional Dataset: UNSW-NB15 (Network-Based IoT Scenario)

In order to overcome the limitation of single-dataset evaluation, an additional dataset was incorporated into the experimental analysis. The UNSW-NB15 dataset was selected as it represents network-based IoT environments, particularly in intrusion detection and cybersecurity scenarios [42, 43].

This dataset contains network traffic features describing both normal and malicious activities. The original dataset contains 257,673 network traffic instances, divided into predefined training and testing subsets. It includes numerical and categorical variables, which were preprocessed using one-hot encoding to ensure compatibility with the evaluated models.

Compared to the HAR dataset, which reflects sensor-based IoT applications, UNSW-NB15 provides a complementary perspective by focusing on network-level data. This dual evaluation enables a more comprehensive assessment of model behavior across heterogeneous IoT contexts.

C. Experimental Protocol

Five supervised classification algorithms were evaluated, in consistency with those analyzed in the initial conference paper presented at COCIA’2025 [13]:

- Naïve Bayes (NB).
- Decision Tree (DT).
- Random Forest (RF).
- k-Nearest Neighbors (k-NN).
- Support Vector Machine (SVM).

For both datasets, the same experimental protocol was applied to ensure consistency and fairness. Feature normalization and one-hot encoding were used where necessary. The models were evaluated using both predictive metrics (accuracy, precision, recall, F1-Score) and computational metrics (training time and inference time), in order to reflect real deployment constraints in IoT and edge environments.

Performance evaluation relied on both predictive and computational metrics, in line with the constraints of IoT systems:

- Overall accuracy.
- Macro-averaged precision.
- Macro-averaged recall.
- Macro-averaged F1-Score.
- Training time (in seconds).
- Inference time (in seconds).

Inference time is particularly critical in IoT and edge computing environments, where real-time prediction and low latency are required.

This evaluation protocol was designed to reflect practical deployment considerations, rather than focusing exclusively on predictive accuracy.

The hyperparameter configuration used for each evaluated classifier is summarized in Table II.

TABLE II. HYPERPARAMETER CONFIGURATION OF EVALUATED MODELS

Model	Hyperparameters
NB	Default (GaussianNB)
DT	criterion = gini, max_depth = None, random_state = 42
RF	HAR dataset: *n*_estimators = 100, random_state = 42, *n*_jobs = −1 UNSW-NB15 dataset : *n*_estimators = 50, random_state = 42, *n*_jobs = −1
k-NN	*n*_neighbors = 5
SVM	HAR dataset : kerne l = "rbf", gamma = scale UNSW-NB15 dataset : kernel = linear (subset training), gamma = scale

The models were configured using standard hyperparameter settings commonly adopted in the literature, without extensive tuning, in order to ensure a fair and consistent comparison across algorithms.

Cross-validation was not applied in this study, as the objective was to evaluate model behavior under a fixed and reproducible experimental setup, consistent with deployment-oriented evaluation scenarios. This ensures reproducibility of the experimental setup.

All experiments were conducted using Python and the Scikit-learn machine learning library within the Kaggle cloud-based computing environment. Identical preprocessing, training, and evaluation procedures were applied across all models to ensure consistency and reproducibility of the results.

D. *Experimental Results*

Table III reports the experimental performance of the evaluated classifiers on the HAR dataset.

TABLE III. EXPERIMENTAL PERFORMANCE ON THE HAR DATASET

Model	Accuracy	Precision	Recall	F1-Score	Training Time (s)
NB	0.7955	0.8080	0.7966	0.7923	0.1024
DT	0.9324	0.9313	0.9312	0.9312	9.2007
RF	**0.9786**	**0.9785**	**0.9786**	**0.9786**	7.4347
k-NN	0.9657	0.9676	0.9675	0.9675	**0.0230**
SVM	0.9693	0.9707	0.9706	0.9706	2.6413

The best results for each metric are highlighted. As expected, notable differences can be observed across models in terms of both predictive performance and computational efficiency.

To further validate the proposed framework and improve generalizability, additional experiments were conducted on the UNSW-NB15 dataset.

TABLE IV. EXPERIMENTAL PERFORMANCE ON THE UNSW-NB15 DATASET

Model	Accuracy	Precision	Recall	F1–Score	Training Time (s)	Inference Time (s)
NB	0.5559	0.7514	0.5967	0.4967	2.3541	0.7370
DT	0.8636	0.8786	0.8531	0.8585	6.5017	**0.5450**
RF	**0.8721**	**0.8967**	**0.8595**	**0.8661**	8.4161	0.7795
k-NN	0.8446	0.8690	0.8310	0.8368	**1.8433**	114.2717
SVM (subset)	0.8106	0.8712	0.7894	0.7933	23.1412	18.4613

The analysis of the UNSW-NB15 results highlights specific characteristics of network-based IoT data. Random Forest demonstrates strong performance in handling high-dimensional and heterogeneous traffic features, while Decision Tree provides a good balance between accuracy and computational efficiency. In contrast, k-NN, despite acceptable predictive performance, exhibits significantly higher inference time, making it less suitable for real-time network monitoring scenarios. These results emphasize the importance of considering both predictive and computational aspects when selecting models for large-scale IoT systems.

These observations further validate the relevance of the proposed multi-objective evaluation framework in large-scale IoT scenarios.

E. *Analysis and Interpretation of Results*

The experimental results highlight clear performance disparities among the evaluated algorithms. Random Forest achieves the highest overall performance in terms of accuracy, recall, and F1-Score, confirming its ability to handle high-dimensional sensor data and complex decision boundaries, which are typical of IoT sensing scenarios. These observations are consistent with the trends identified in the UNSW-NB15 dataset.

Fig. 2 (HAR Dataset) illustrates the balance between predictive performance and computational efficiency, highlighting the multi-objective nature of algorithm selection in IoT and edge computing environments.

Fig. 2 provides a visual representation of the trade-off between predictive performance and computational cost by plotting overall accuracy against training time for the evaluated classifiers. The results clearly illustrate that algorithms achieving the highest accuracy, such as Random Forest and Support Vector Machine, are associated with higher computational costs, making them more suitable for edge or cloud-based deployment.

Conversely, lightweight models such as Naïve Bayes and k-NN exhibit significantly lower training times, which makes them attractive for resource-constrained IoT nodes, despite a moderate reduction in predictive performance. This visualization confirms the relevance of a multi-objective perspective for algorithm selection in connected systems.

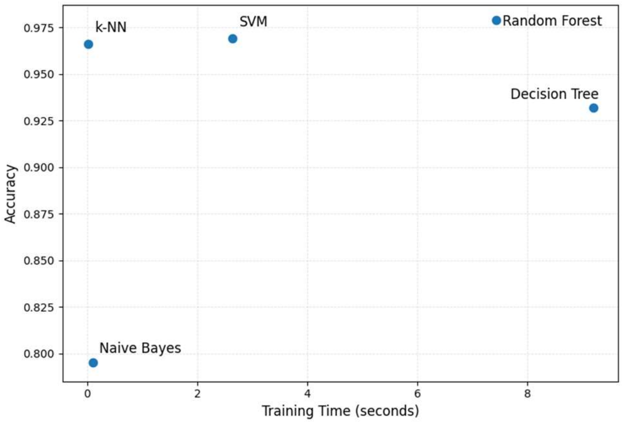

Fig. 2. Trade-off between accuracy and training time for the evaluated classification algorithms on the HAR dataset.

Support Vector Machine also achieves strong predictive performance, while k-NN offers a favorable compromise between accuracy and training efficiency. In particular, k-NN exhibits an extremely low training time, which makes it attractive for scenarios requiring rapid model updates or execution at the network edge. However, its reliance on distance computations may introduce scalability limitations during inference, resulting in significantly higher inference time in large-scale scenarios.

Decision Tree offers an interesting compromise between predictive performance and interpretability, which is a critical requirement in domains such as healthcare or human-centered IoT applications. Nevertheless, its training time is higher than that of simpler models in this experimental setting.

Finally, Naïve Bayes, while exhibiting lower predictive performance, remains highly relevant for resource-constrained IoT nodes, due to its minimal training time and low computational overhead. This confirms its suitability for embedded or battery-powered devices where efficiency is paramount. This highlights the relevance of lightweight models in energy-constrained IoT environments.

While these observations are derived from the HAR dataset, further validation on additional datasets is necessary to confirm the generalizability of these findings across different IoT contexts.

The computational complexity of the evaluated models varies significantly. Naïve Bayes has linear complexity $O(n)$, making it highly efficient for both training and inference. Decision Tree and Random Forest exhibit higher complexity due to tree construction and ensemble learning, typically $O(n \log n)$ or higher depending on implementation.

k-NN has low training cost but high inference complexity $O(n)$, as it requires distance computation with all training samples, which explains its high latency observed in the experiments.

SVM presents higher computational complexity, particularly for large-scale datasets, which justifies the use of a subset for practical evaluation.

Although energy consumption was not directly measured, the observed computational complexity and inference latency provide indirect indicators of deployment-related energy requirements in IoT environments.

F. Consistency with the Multi-Objective Evaluation Framework

These experimental findings are fully consistent with the theoretical multi-objective framework introduced in Section III and the comparative analysis presented in Section IV. As anticipated, no single algorithm dominates all evaluation criteria, reinforcing the necessity of a trade-off-based selection strategy.

Algorithms achieving the highest predictive performance, such as Random Forest and SVM, incur higher computational costs, making them more suitable for edge or cloud-based deployment. In contrast, lighter models such as Naïve Bayes and Decision Tree are better aligned with IoT nodes operating under strict resource constraints, despite their lower accuracy.

Overall, this experimental validation confirms that the selection of classification algorithms in connected systems should be guided by application-specific constraints and deployment architecture, rather than by predictive performance alone. These results further support the relevance and practical applicability of the proposed multi-objective evaluation framework.

A comparative analysis across the two datasets reveals consistent patterns in model behavior, confirming the robustness of the proposed multi-objective framework. The inclusion of both sensor-based and network-based datasets significantly improves the generalizability of the evaluation. This further confirms the robustness and applicability of the proposed framework across diverse IoT scenarios.

VI. Discussion

This section discusses the main lessons learned from theoretical analysis and experimental validation, placing them in the context of the actual deployment of classification algorithms in connected systems and IoT architectures.

A. Discussion of Multi-objective Trade-offs

The results obtained confirm that the selection of a classification algorithm in IoT environments cannot be reduced to a single predictive performance criterion. Theoretical (Section IV) and experimental (Section V) analyses clearly show that algorithms offering the best performance in terms of accuracy, such as Random Forest and Support Vector Machine, come with higher computational costs, both in the learning phase and in the inference phase.

Conversely, simpler models such as Naïve Bayes or Decision Tree have more modest predictive performance, but offer significant advantages in terms of simplicity, speed of execution, and resource consumption. These observations clearly illustrate the conflicting nature of the objectives considered and fully justify the adoption of a multi-objective approach, such as the one formalized in this work and initiated in our initial contribution presented at COCIA'2025 [13]. In addition, although energy consumption was not directly measured, inference time and computational cost were used as practical proxies for evaluating deployment constraints in IoT environments.

Although inference time was not originally reported for the HAR dataset, it was explicitly included in the extended UNSW-NB15 evaluation, where latency becomes more critical due to the larger scale and network-oriented nature of the data. This provides a practical proxy for deployment-oriented computational analysis.

The inclusion of the UNSW-NB15 dataset significantly improves the generalizability of the results by covering both sensor-based and network-based IoT scenarios.

These findings are consistent across both HAR and UNSW-NB15 datasets, reinforcing the robustness of the observed trade-offs.

B. Implications for Deployment in IoT Architectures

The results enable practical recommendations to be made for the deployment of classification algorithms according to the target architecture. On highly constrained IoT nodes, characterized by limited energy, memory, and computing resources, as well as strict latency constraints highlighted by inference time analysis, lightweight and inexpensive models, such as Naïve Bayes or simple decision trees, appear to be relevant choices [44].

In edge computing architectures, where a compromise between latency and performance is sought, algorithms such as k-NN, Decision Tree, or optimized versions of SVM may be suitable solutions, depending on the specific constraints of the application [45]. Finally, in cloud environments, where resources are more abundant, more complex models such as Random Forest or SVM may be preferred in order to maximize accuracy and robustness.

This deployment typology illustrates the practical value of the proposed evaluation framework, which aims to guide the choice of algorithms based on actual constraints rather than providing a universal ranking. This further emphasizes the importance of multi-objective decision-making strategies in real-world IoT deployments.

C. Limitations of the Study

Despite the encouraging results obtained, certain limitations must be highlighted. The experimental analysis is based on two complementary datasets representing different IoT contexts (sensor-based and network-based). Although the use of independent evaluation data improves generalizability of the findings, further validation across additional datasets and real-world deployment settings would enhance the robustness of the conclusions [46].

Furthermore, the parameters of the evaluated algorithms were chosen in a standard manner, without in-depth optimization. However, fine adjustments to hyperparameters can significantly influence model performance and computational requirements, since hyperparameter configurations affect prediction accuracy, convergence dynamics, training efficiency, and resource utilization [47, 48]. These limitations open the way for prospects for extending and deepening the proposed framework.

VII. CONCLUSION

In this article, we have proposed a substantial extension of our initial work presented at the COCIA'2025 conference, introducing a multi-objective evaluation framework for selecting supervised classification algorithms in connected systems and IoT environments. Unlike traditional approaches focused solely on predictive performance, the proposed framework simultaneously integrates criteria of accuracy, robustness, computational complexity, and deployment feasibility, including latency-related constraints reflected by inference time.

A comparative theoretical analysis, based on previous results, highlighted the structural trade-offs between the main classification algorithms. This analysis was supplemented by experimental validation on two complementary datasets, namely the HAR dataset (sensor-based IoT) and the UNSW-NB15 dataset (network-based IoT), representing both sensor-based and network-based IoT scenarios. The experimental results confirm that no single algorithm dominates all of the criteria considered, and that the choice of model must be guided by the application context and deployment architecture. The inclusion of both datasets improves the generalizability of the results across heterogeneous IoT scenarios. This confirms the relevance of a multi-objective approach for real-world IoT deployments. These findings provide practical guidance for selecting classification models under real-world IoT constraints.

Future work will focus on several areas. First, extending the experimental framework to other IoT datasets, particularly those from industrial contexts, environmental sensor networks, or smart monitoring systems, will strengthen the generality of the conclusions. Second, the integration of additional metrics, such as detailed energy consumption measurements and deployment experiments on embedded hardware platforms, would further improve the practical relevance of the framework. Finally, the exploration of automatic optimization methods, such as Automated Machine Learning (AutoML) or dynamic model selection, opens up interesting prospects for the online adaptation of classification algorithms in scalable IoT systems. This paves the way for more adaptive and intelligent model selection strategies in future IoT systems.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

Mouataz Idrissi Khaldi contributed to the conceptualization of the study, methodology development, experimental design and implementation, data analysis, visualization, and preparation of the original manuscript; Allae Erraissi contributed to the methodology, analysis, validation of the experimental approach, interpretation of the results, and review and editing of the manuscript; Mustapha Hain contributed to the literature analysis, methodological validation, interpretation and verification of the results, and critical review and revision of the manuscript; Mouad Banane contributed to the conceptual development of the study, supervision, validation of the overall research framework, interpretation of the findings, and critical review and editing of the manuscript. All authors discussed the results, contributed to the revision of the manuscript, and approved the final version.

REFERENCES

[1] X. Zhu *et al.*, "Machine learning applications: A survey of current trends and future directions," *IEEE Trans. on Knowledge and Data Engineering*, vol. 36, no. 1, pp. 45–67, 2023.

[2] N. A. Ange, D. Ravindra, P. Vincent, K. Srinivasa, and Y. C. Hu, "Recent advances in evolving computing paradigms: Cloud, edge, and fog technologies," *Sensors*, vol. 22, no. 1, 196, 2022

[3] M. P. Véstias, R. P. Duarte, J. T. de Sousa, and H. C. Neto, "Moving deep learning to the edge," *Algorithms*, vol. 13, no. 5, 125, 2020.

[4] O. Jouini, K. Sethom, A. Namoun, N. Aljohani, M. H. Alanazi, and M. N. Alanazi, "A survey of machine learning in edge computing: Techniques, frameworks, applications, issues, and research directions," *Technologies*, vol. 12, no. 6, 81, 2024

[5] W. Shi, J. Cao, Q. Zhang, Y. Li, and L. Xu, "Edge computing: Vision and challenges," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, 2016.

[6] E. Ramanujam, T. Perumal, and S. Padmavathi, "Human activity recognition with smartphone and wearable sensors using deep learning techniques: A review," *IEEE Sensors J.*, vol. 21, no. 12, pp. 13029–13040, 2021.

[7] H. Khalaf and M. Riyadh, "Human activity recognition using inertial sensors in a smartphone: Technical background (Review)," *Al-Nahrain J. Sci.*, vol. 27, no. 1, pp. 108–120, 2024.

[8] S. Raschka and V. Mirjalili, *Python Machine Learning: Machine Learning and Deep Learning with Python, Scikit-Learn, and Tensor Flow 2*, 3rd ed. Birmingham, U.K.: Packt Publishing, 2019.

[9] E. O. Owusu *et al.*, "A systematic review of machine learning and deep learning techniques for DDoS attack mitigation in IoT and edge computing systems," *Security Privacy*, vol. 9, no. 1, e70147, 2026.

[10] S. M. Raza, J. Jeong, M. Kim, B. Kang, and H. Choo, "Empirical performance and energy consumption evaluation of container solutions on resource constrained IoT gateways," *Sensors*, vol. 21, no. 4, 1378, 2021.

[11] C. M. Bhushan *et al.*, "Deploying TinyML for energy-efficient object detection and communication in low-power edge AI systems," *Sci. Rep.*, vol. 15, 44299, 2025

[12] Y. Mejdoub, A. Elamri, and M. Kardouchi, "Connected objects, artificial intelligence, telecommunications and electronics engineering," in *Proc. COCIA'2025*, vol. 1584, 2026.

[13] M. I. Khaldi, A. Erraissi, M. Hain, and M. Banane, "Benchmarking classification algorithms: An approach based on multi-objective criteria," in *Proc. 3rd Int. Conf. Connected Objects and Artificial Intelligence (COCIA'2025)*, 2026, vol. 1584, pp. 655–662.

[14] A. B. Sada, A. Khelloufi, A. Naouri, H. Ning, and S. Dhelim, "Energy-aware selective inference task offloading for real-time edge computing applications," *IEEE Access*, vol. 12, pp. 72924–72937, 2024

[15] G. A. Araújo, S. F. C. Bezerra, and A. R. Rocha, "Resource allocation based on task priority and resource consumption in edge computing," *Internet Services Appl.*, vol. 15, no. 1, pp. 360–379, 2024.

[16] M. J. C. S. Reis, "Lightweight signal processing and edge AI for real-time anomaly detection in IoT sensor networks," *Sensors*, vol. 25, no. 21, 6629, 2025

[17] V. S. Lalapura, V. R. Bhimavarapu, J. Amudha, and H. S. Satheesh, "A systematic evaluation of recurrent neural network models for edge intelligence and human activity recognition applications," *Algorithms*, vol. 17, no. 3, 104, 2024

[18] C. Zaoui, F. Benabbou, and A. Ettaoufik, "Edge-Fog-Cloud data analysis for eHealth-IoT," *Int. J. Online Biomed. Eng. (iJOE)*, vol. 19, no. 7, pp. 184–199, 2023.

[19] M. A. Mahmood, S. Almuayqil, K. O. Alsalem, and K. Gasmi, "A granular computing classifier for human activity with smartphones," *Appl. Sci.*, vol. 13, no. 2, 1175, 2023.

[20] S. P. John, V. Alappat, N. Varghese, and S. V. Simpson, "Edge-cloud computing for IoT data analytics using optimized progressive graph convolutional networks," *IETE J. Res.*, vol. 71, no. 5, pp. 1744–1751, 2025.

[21] W. Rafique, L. Qi, I. Yaqoob, M. Imran, R. U. Rasool, and W. Dou, "Complementing IoT services through software defined networking and edge computing: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 22, no. 3, pp. 1761–1804, 2020.

[22] M. A. Ja'afreh, H. Adhami, A. E. Alchalabi, M. Hoda, and A. E. Saddik, "Toward integrating software defined networks with the internet of things: A review," *Cluster Comput.*, vol. 25, no. 3, pp. 1619–1636, Jun. 2022.

[23] M. Rostami and S. Goli-Bidgoli, "An overview of QoS-aware load balancing techniques in SDN-based IoT networks," *J. Cloud Comp.*, vol. 13, 89, 2024.

[24] F. Palmese, A. E. C. Redondi, and M. Cesana, "Adaptive quality of service control for MQTT-SN," *Sensors*, vol. 22, no. 22, 8852, 2022.

[25] L. Kong, J. Tan, J. Huang, G. Chen, S. Wang, X. Jin, P. Zeng, M. Khan, and S. K. Das, "Edge-computing-driven Internet of Things: A survey," *ACM Comput. Surv.*, vol. 55, no. 8, 174, 2023.

[26] M. Ouhami, A. Hafiane, Y. E. Saady, M. E. Hajji, and R. Canals, "Computer vision, IoT and data fusion for crop disease detection using machine learning: A survey and ongoing research," *Remote Sens.*, vol. 13, no. 13, 2486, 2021.

[27] A. Sharma, A. Jain, P. Gupta, and V. Chowdary, "Machine learning applications for precision agriculture: A comprehensive review," *IEEE Access*, vol. 9, pp. 4843–4873, 2021.

[28] M. Cruz, S. Mafra, E. Teixeira, and F. Figueiredo, "Smart strawberry farming using edge computing and IoT," *Sensors*, vol. 22, no. 15, 5866, 2022.

[29] D. Marković, Z. Stamenković, B. Đorđević, and S. Ranđić, "Image processing for smart agriculture applications using cloud-fog computing," *Sensors*, vol. 24, no. 18, 5965, 2024.

[30] J. P. Nyakuri *et al.*, "AI and IoT-powered edge device optimized for crop pest and disease detection," *Sci. Rep.*, vol. 15, 22905, 2025.

[31] K. Owa and O. Adewole, "Benchmarking machine learning techniques for phishing detection and secure URL classification," *Int. J. Comput. Sci. Mobile Comput.*, vol. 14, no. 1, pp. 20–37, 2025.

[32] I. Ullah and Q. H. Mahmoud, "A two-level flow-based anomalous activity detection system for IoT networks," *Electronics*, vol. 9, no. 3, 530, 2020.

[33] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: Techniques, datasets and challenges," *Cybersecur.*, vol. 2, 20, 2019.

[34] M. Verkerken, L. Dhooge, T. Wauters, B. Volckaert, and F. D. Turck, "Towards model generalization for intrusion detection: Unsupervised machine learning techniques," *J. Netw. Syst. Manage.*, vol. 30, 12, Jan. 2022.

[35] H. Alkahtani and T. H. H. Aldhyani, "Artificial intelligence algorithms for malware detection in Android-operated mobile devices," *Sensors*, vol. 22, no. 6, 2268, 2022.

[36] M. D. Mauro, G. Galatro, G. Fortino, and A. Liotta, "Supervised feature selection techniques in network intrusion detection: A critical review," *Eng. Appl. Artif. Intell.*, vol. 101, 104216, 2021.

[37] K. Walji, A. Erraissi, A. Zakrani, and M. Banane, "Sentiment analysis revisited: A multi-metric comparative study," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 16, no. 9, pp. 1–9, 2025.

[38] Q. Renau and E. Hart, "Algorithm selection with probing trajectories: Benchmarking the choice of classifier model," in *Proc. International Conference on the Applications of Evolutionary Computation*, 2025, pp. 452–468.

[39] K. Walji, A. Erraissi, A. Zakrani, and M. Banane, "From review to practice: A comparative study and decision-support framework for sentiment classification models," *Int. J. Adv. Comput. Sci. Appl. (IJACSA)*, vol. 16, no. 9, pp. 699–709, 2025.

[40] K. Deb, *Multi-Objective Optimization Using Evolutionary Algorithms*, New York, NY, USA: Wiley, 2001.

[41] D. Anguita, A. Ghio, L. Oneto, X. Parra, and J. L. R. Ortiz, "A public domain dataset for human activity recognition using smartphones," in *Proc. 21st Eur. Symp. Artif. Neural Netw., Comput. Intell. Mach. Learn. (ESANN)*, 2013, pp. 1–6.

[42] N. Moustafa and J. Slay, "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," in *Proc. Mil. Commun. Inf. Syst. Conf. (MilCIS)*, 2015, pp. 1–6.

[43] N. Moustafa and J. Slay, "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 dataset and the comparison with the KDD99 dataset," *Inf. Secur. J.: A Global Perspective*, vol. 25, no. 1–3, pp. 18–31, 2016.

[44] M. R. Rahman, M. Rahman, I. Rasul, M. H. Arif, M. A. Alim, M. S. Hossen, and T. Bhuiyan, "Lightweight machine learning models for real-time ransomware detection on resource-constrained devices," *J. Inf. Commun. Technol. Robot. Appl.*, vol. 15, no. 1, pp. 17–23, 2025.

[45] J. Verbraeken, M. Wolting, J. Katzy, J. Kloppenburg, T. Verbelen, and J. S. Rellermeyer, "A survey on distributed machine learning," *ACM Comput. Surv.*, vol. 53, no. 2, 30, 2021.

[46] F. Cabitza, A. Campagner, F. Soares, L. G. G. Romualdo, F. Challa, A. Sulejmani, M. Seghezzi, and A. Carobene, "The importance of being external: Methodological insights for the external validation of machine learning models in medicine," *Comput. Methods Programs Biomed.*, vol. 208, 106288, 2021.

[47] F. Seifi and S. T. A. Niaki, "Extending the hypergradient descent technique to reduce the time of optimal solution achieved in hyperparameter optimization algorithms," *Int. J. Ind. Eng. Comput.*, vol. 14, no. 3, pp. 501–510, 2023.

[48] R. Narayanan and N. Ganesh, "A comprehensive review of metaheuristics for hyperparameter optimization in machine learning," in *Metaheuristics for Machine Learning: Algorithms and Applications*, K. Kalita, N. Ganesh, and S. Balamurugan, Eds. Hoboken, NJ, USA: Wiley, 2024, pp. 37−72.