International Transactions on Electrical Engineering and Computer Science
A. K. Alhossainy et.al., Vol: 5, No: 1, pp: 1-16, March 2026

AI Powered Threat Detection Framework for Security Enhancement of MQTT based IoT Networks

Ali Kareem Alhossainy[1], Mina Malekzadeh[1]

Abstract: The Message Queuing Telemetry Transport (MQTT) protocol is widely adopted in IoT networks due to its lightweight and scalable design, making it ideal for resource-constrained devices. However, its minimal architecture lacks built-in encryption and robust identity authentication, leaving it vulnerable to a range of security threats. To address these vulnerabilities, this work proposes a hybrid framework that integrates machine learning (ML) and deep learning (DL) models to enhance intrusion and anomaly detection in MQTT based IoT networks. The framework leverages real world IoT traffic data from the MQTTEEB-D dataset with diverse preprocessing techniques to train the model. The framework is implemented and evaluated through a multi-metric approach, where each metric assesses a distinct aspect of the framework to identify lightweight processing models for resource-constrained environments. Results consistently showed that ML models outperformed their DL counterparts in terms of detection reliability, classification balance, and error minimization. While DL models demonstrated moderate effectiveness in capturing temporal patterns, they exhibited higher misclassification rates and reduced calibration. The findings underscore the effectiveness of lightweight ML models for scalable and dependable intrusion detection in MQTT based IoT networks.

Keywords: MQTT, IoT Security, Machine Learning, Deep Learning, Intrusion Detection.

1. Introduction

The Internet of Things (IoT) enables a vast array of devices, from simple sensors to complex industrial machines, to connect and form a ubiquitous network that continuously senses, processes, and shares data [1]. This interconnectivity has sparked a wide range of innovations in automation, productivity, and real time decision-making systems.

History
Received: 09-11-2025; Revised: 16-12-2025;
Accepted: 18-12-2025

✉ Mina Malekzadeh
m.malekzadeh@hsu.ac.ir

[1]Electrical and Computer Engineering Faculty, Hakim Sabzevari University, Sabzevar, 9617976487, Iran.

However, because these devices are often resource-constrained, such as limited power, memory, and processing capacity [2], IoT has necessitated the development of specialized communication protocols to meet these demands. One widely adopted protocol in IoT communication is Message Queuing Telemetry Transport (MQTT), a lightweight, scalable, and efficient messaging protocol designed for low-bandwidth networks. MQTT's publish-subscribe architecture is particularly well-suited for IoT applications, enabling reliable device-to-device communication with minimal overhead. This makes MQTT especially attractive in resource-constrained environments like sensor networks and embedded systems [3]. Despite its efficiency and scalability, MQTT suffers from notable security limitations. The protocol's current security mechanisms, typically basic username and password authentication, are insufficient to address the dynamic and evolving threats faced by IoT networks. These static authentication methods are vulnerable to a variety of

https://doi.org/10.62760/iteecs.5.1.2026.171

attacks, including unauthorized access, eavesdropping, man-in-the-middle attacks, and data tampering. As IoT networks continue to expand and grow in complexity, traditional security practices become inadequate for protecting sensitive data and ensuring the integrity of communications [4]. To respond to these problems, the application of Artificial Intelligence (AI) in IoT security solutions has been more popular in recent years. AI learning algorithms offer dynamic, adaptive solutions that are capable of learning from network traffic patterns on a continuous basis and evolving according to new threats. With the capability of anomaly detection, behavior-based authentication, and intelligent decision-making algorithms, AI-based security solutions can provide real-time protection against emerging threats without relying on static, pre-programmed rules while adapting to variations in IoT networks. This shift from static to dynamic security mechanisms can potentially make IoT networks more secure and resilient [5].

Despite its promise, applying AI to protocols like MQTT presents several notable challenges. Most IoT devices operate with limited processing power, memory, and bandwidth, making it difficult to deploy computationally intensive AI algorithms without compromising performance. Moreover, IoT networks are highly heterogeneous, and small datasets often lack the diversity needed for effective generalization. Consequently, training robust AI models requires large, diverse datasets, posing another challenge for resource-constrained IoT networks [6]. Furthermore, the AI domain encompasses a broad spectrum of ML and DL techniques, each with unique strengths and limitations when applied to IoT systems. Selecting and customizing the most suitable approach for a given IoT network is a complex task, especially when the focus is on environments with scarce computational resources.

To address these challenges, this work proposes a hybrid framework that integrates ML and DL models to enhance intrusion and anomaly detection in MQTT-based communication while maintaining system performance. Accordingly, the main contributions of this work can be summarized as follows:

- ***Hybrid AI-based framework:*** A hybrid intrusion and anomaly detection framework is proposed that integrates both machine learning and deep learning models. A diverse set of ML algorithms (RF, GB, AdaBoost, LR, KNN, DT, NB, SVM, and K-Means) and DL architectures (RNN, GRU, BiLSTM, DNN, LSTM, and CNN) are systematically assessed to capture their complementary strengths. ML provides lightweight efficiency suitable for constrained devices, while DL captures complex temporal and nonlinear patterns. This dual evaluation offers a more balanced and thorough understanding of security solutions for MQTT-based IoT networks.
- ***Framework optimization:*** To ensure robust performance and fair comparison across models, a grid search algorithm is employed to optimize hyperparameters, enhancing detection accuracy and efficiency in resource-constrained IoT environments.
- ***Identification of efficient solutions:*** Through extensive experimental evaluation, the framework is evaluated using multiple performance measures, allowing a balanced assessment that reflects overall reliability rather than relying on a single indicator.

The rest of this work is organized as follows: Section 2 reviews related works, Section 3 details the methodology for developing and implementing the framework, Section 4 presents the results of the framework implementation, and Section 5 concludes the work.

2. Related Works

Since security in the IoT networks has always been an issue, different methods have been presented to provide secure communication [7 - 8]. AI-based learning methods are widely used to detect malicious traffic patterns and identify various attacks in complex networks, as they effectively extract diverse features from large datasets. Gadze et al. presented employing deep learning models, including Long Short Term Memory (LSTM) and Convolution Neural Network (CNN), to identify DDoS intrusions within a software-defined network's centralized controller. The results showed that the LSTM model achieved an accuracy rate of 89.63%, while the CNN model achieved a lower accuracy rate of 66%. One of the limitations of their study was that the DDoS intrusion detection was time-consuming, especially with LSTM models. The present study overcomes this limitation by optimizing the time efficiency of the LSTM model and searching for

more time-effective models for real-time intrusion detection [9]. Ahuja et al. created a hybrid machine learning model based on Support Vector Machines (SVM) and random forest (SVC-RF) to distinguish between malicious and normal traffic. They extracted features from the raw data to create an SDN dataset of new features. The result was that the SVC-RF classifier could achieve a high accuracy of 98.8% on the SDN dataset. The limitation of this approach is its dependence on the quality of the feature extraction and having an adequately tagged dataset. This limitation is tackled by the present study via the addition of automated feature selection and improvement techniques to promote model generalization [10].

Wang et al. introduced a new deep learning model based on an improved deep belief network (DBN) with a kernel-based extreme learning machine (KELM) applied to enhance network intrusion speed. The results of experiments showed that the DBN-KELM model achieved 93.5% accuracy, and the DBN-EGWO-KELM approach improved this to 98.6%. A shortcoming of this model was that it relied on the kernel-based learning mechanism, which was still in its early stages of development. The present work addresses this deficiency by offering more mature and stable approaches to intrusion detection while maintaining Wang et al.'s high rate of accuracy [11]. Dotcenko et al. introduce an SVM-based method for DDoS attack detection in SDNs. The average accuracy rate was 95.24% which confirms the effectiveness of the method to identify the DDoS attacks. The study utilized a limited 6-tuple feature to detect the DDoS attacks, and the training-test data ratio was not specified, which limits the experimental design clarity. For future research, a larger feature set could be explored, and also an articulated training-test split so that transparency and the robustness of the solution for various attack scenarios can be improved [12].

Abhiroop et al. suggest the use of Neural Networks, Naïve Bayes, and SVM for the detection of DoS attacks in SDN switches. The result shows Naïve Bayes and Neural Networks were 100% correct, and SVM was 99% correct. The dataset was captured using a 60/40 train-to-test ratio typical in machine learning experiments. One limitation of the study is that it addresses the detection of DDoS attacks solely in the data plane. Future research can involve the comparison of the approach with a wider variety of other types of attacks and networks, providing a more universal solution for the detection of DoS attacks in SDNs [13]. Sahoo et al. compare some machine learning models, including Linear Regression (LR), Naïve Bayes, KNN, Decision Tree, Random Forest (RF), SVM, and Artificial Neural Networks (ANN), for the prediction of DDoS traffic in SDNs. The precision is found to be maximum for Linear Regression at 98.65% and minimum for Naïve Bayes at 97.45%. The experiment was performed with different train-test splits (90/10, 80/20, and 70/30), but on UDP and ICMP flood attacks alone. One limitation of the study is that it concentrated mostly on some specific types of attacks [14].

In addition to AI-based models, some researchers focus on the traditional methods for secure communication in IoT networks. Carlier et al. propose a symmetric key based scheme for wireless sensor network multicast communication security. This scheme is significant in offering confidentiality, integrity, and mutual authentication among nodes of a multicast group. The authors highlight the fact that their approach is robust in scenarios where trust between participants can be guaranteed. But scalability for this method is a problem when applied to large networks since it has more communication overhead and computational expense of dealing with keys in numerous nodes. The authors suggest that the future solution can involve distributed or hierarchical designs in dealing with keys so that the system will be scalable enough while still preserving the security properties [15].

Shabisha et al. introduce a key management scheme to facilitate secure group communication among IoT devices and fog devices based on Lagrange interpolation and elliptic curve cryptography. It shows how the technique facilitates secure group key distribution and communication that can be applied in fog computing environments. However, the computational overhead of elliptic curve cryptography, especially when applied to resource-constrained IoT devices, makes the approach inefficient in the context of large-scale or low-power IoT networks. The authors propose enhancing cryptographic performance by studying other key management schemes that involve less computation without compromising security [16].

Chandramouli et al. explain well-protected verifiable secret-sharing schemes with emphasis on

schemes robust against adversaries of unbounded computational power. The paper explains quantum cryptography based secret-sharing schemes based on multi dimensional threshold secret sharing, which has theoretically high security. However, the biggest disadvantage of such protocols is the practicality in group key generation, which implies the simultaneous interaction of all the participants to combine the secrets, and hence, is not practicable for use in real scenarios. The authors draw the conclusion that the future work should focus on the identification of more practical and efficient substitutes for group key generation that would be usable in less-than-ideal settings [17]. Tiloca et al. suggested the Axiom protocol as a means of securing unicast communication within multicast groups. Through the creation of individual keys from an underlying group key, the protocol attempts to secure messages with a form of centralized key management. But one of the greatest disadvantages of this approach is that it doesn't solve the authentication problems well because any group member can compute the group key. This poor authentication can lead to potential security risks in situations where group members do not fully trust each other. The authors suggest that adding a trusted third-party for key management could overcome this restriction and also increase the security of the system [18].

In [19] authors discuss the challenge of imbalanced datasets in IoT intrusion detection, where attack traffic is underrepresented compared to normal traffic. To detect attacks on MQTT protocol, they introduce three hybrid IDSs, including CNN-RNN, CNN-LSTM, and CNN-GRU, by leveraging Generative Adversarial Networks (GANs) to create balanced datasets. The results highlight the effectiveness of the GAN-based approach in improving detection accuracy. However, the work does not provide a comparative analysis with broader DL variants or ML methods to identify lightweight solutions suitable for resource-constrained IoT devices. In [20], it is noted that since MQTT was designed for lightweight communication, it lacks robust security mechanisms, leaving it susceptible to diverse cyberattacks. To enhance the performance of MQTT-based systems, the authors introduced an IDS employing seven machine learning models, including SGD, LR, RF, DT, NB, k-NN, and XGBoost, for anomaly detection and traffic classification. While the results highlight the superior performance of the DT model, the study's contribution remained focused on ML based approaches, with alternative deep learning methods not considered within its scope. Vulnerabilities of MQTT enabled networks to various security threats, including eavesdropping, weak authentication, and malicious payload injections, are discussed in [21]. The study further notes that detecting such intrusions is particularly challenging in IoT environments due to constrained resources, high traffic volumes, and heterogeneous attack vectors. To address these challenges, an ensemble learning-based IDS is proposed, integrating Gaussian Naive Bayes (GNB), Kernel based SVM (K-SVM), and Multi Layer Perceptron (MLP) as base classifiers, with LR serving as the meta-classifier. The results highlight the stronger performance of MLP within the ensemble. The contribution, however, remains focused on this ensemble framework, without extending evaluation to a broader range of ML or DL methods that could provide additional comparative insights. The authors in [22] consider MQTT as a popular IoT application layer protocol, and highlight its vulnerability to various attacks such as Basic Connect Flooding, Delay Connect Flooding, Invalid Subscription Flooding, Connect Flooding with WILL Payload and TCP SYN Flooding exploitation. A detection approach is proposed to identify anomalies in the MQTT communication sequence to detect anomalous requests. Hybrid DL models including CNN-RNN, CNN-LSTM, and CNN-GRU are unutilized in [23], to enhance MQTT security against cyber-attacks. However, the ML models are not implemented to provide a basis evaluation for resource constraint IoT network, leaving their practical applicability unverified.

Despite the promising results achieved by existing AI based and traditional security approaches, their applicability to MQTT based IoT networks remains limited. Many of the reviewed models focus predominantly on SDN architectures and specific attack types such as DDoS or UDP floods, leaving a gap in addressing the lightweight, asynchronous, and resource-constrained nature of the MQTT protocol. Furthermore, some solutions rely on manual feature engineering or assume trust among nodes, which reduces the robustness and generalizability of these solutions. Additionally, traditional cryptographic schemes, while secure, often impose computational

burdens unsuitable for low power IoT devices. These gaps underscore the need for a more robust, efficient, and scalable solution. The proposed AI-based framework attempts to address these limitations by integrating a diverse set of machine learning and deep learning models. Through a multi-metric evaluation approach, where each metric assesses a distinct aspect of the framework, lightweight processing models are selected to ensure robust security for resource-constrained MQTT-based IoT networks.

3. Methodology

To enhance intrusion and anomaly detection in MQTT - based IoT networks, this work proposes a hybrid framework that integrates both ML and DL models. The methodology encompasses three key stages: data acquisition and preparation, model selection and optimization, and model evaluation, as shown in Fig. 1.

3.1 Data Acquisition and Preparation

As a real IoT deployment, the MQTTEEB-D dataset [24] contains live data of MySignals IoT health sensors, Raspberry Pi 4 development boards, and an MQTT broker server. In contrast to simulation datasets, it depicts the complexities of the IoT communication in the real world under different attacks, including MQTT publish flooding. The information was recorded with PyShark and cut into CSV files so that the traffic of the network and intrusion detection models can be studied in detail. The dataset provides multiple forms of the data. Accordingly, this work selects Raw_RealTime_Data from MQTTEEB-D_Final_Dataset, to implement custom corresponding preprocessing required by each model, which makes them compatible with the models before training. There are approximately 137320 records of MQTT traffic, covering both benign and malicious communication patterns, with 13 features extracted from MQTT traffic flows. Following dataset collection, we conduct a structured preprocessing phase to enhance compatibility and learning efficiency. Several preprocessing techniques are applied. First, the feature selection process is performed, which is a crucial step in the data preprocessing pipeline to determine which columns of the data are redundant or irrelevant to be removed. Such features can negatively affect model performance, increase computational complexity, and contribute to overfitting. In this context, as illustrated in the following code snippet, features are selected from multiple layers of the network to capture diverse behavioral patterns and operational characteristics.

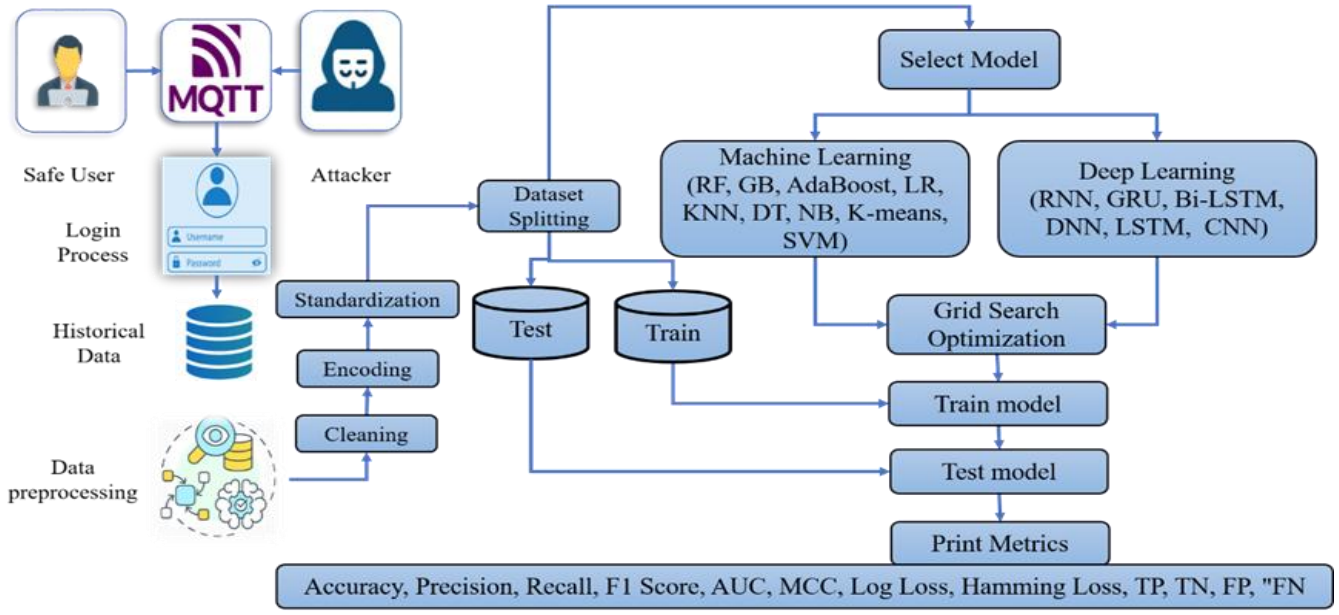

Fig. 1: Proposed framework for anomaly detection

```
relevant_columns = ['timestamp','tcp_flags','tcp_time_delta','tcp_len',
                    'mqtt_conack_flags','mqtt_conflag_cleansess','mqtt_conflags',
                    'mqtt_dupflag','mqtt_hdrflags','mqtt_kalive','mqtt_msg',
                    'mqtt_qos','label']

X = df.drop(columns=['label'])
y = df['label']
```

In addition, raw data can vary significantly in scale, which can affect the performance of learning models that are sensitive to the scale of the data. Without proper scaling, features with larger ranges can influence the learning process of the model. Therefore, we use Standard Scaler from scikit-learn, which transforms each feature to have a mean of 0 and a standard deviation of 1 (within the range of [0.0, 1.0]). This allows each feature to contribute equally to the learning process. Then, to further enhance the efficiency of the model and reduce computational cost, Principal Component Analysis (PCA) is employed as a dimensionality reduction technique. PCA transforms the original feature space into a smaller set of components that capture the highest variance in the data. In this work, the dataset is reduced to two principal components (PCA (n_components=2)), enabling easier visualization while retaining the most informative aspects of the data. This reduction significantly decreases training time and computational complexity without substantial loss of information. As the last step in data preprocessing, we perform class balancing. Because of the class imbalance between normal and malicious samples, the Synthetic Minority Oversampling Technique (SMOTE) is used to generate additional synthetic minority samples, improve model generalization, and reduce bias toward the majority class. It is applied after the train/test split to avoid data leakage. The number of minority class neighbors to use when generating a synthetic sample is 5, and then the classes are balanced to the majority class (SMOTE (n_neighbors=5, sampling_strategy='auto', random_state=42)). Finally, the dataset is split into training (80%) and testing (20%) subsets to support generalization and reduce computational overhead during evaluation.

3.2 Model Selection and Optimization

Multiple models are selected from both ML and DL algorithms to leverage their complementary strengths for intrusion detection on the MQTTEEB-D dataset. The ML models offer fast, interpretable decisions, while DL models excel at capturing complex temporal patterns. This diversity enables a balanced evaluation across detection accuracy, scalability, and adaptability to varied MQTT attack patterns. The selected ML models include RF, GB, AdaBoost, LR, KNN, DT, NB, SVM, and K-Means. RF is an ensemble method that mitigates overfitting by averaging multiple decision trees, enhancing robustness and accuracy. GB builds sequential trees that correct predecessor errors, making it adept at uncovering complex patterns. AdaBoost emphasizes misclassified instances, improving performance on noisy or imbalanced data. LR serves as a reliable baseline for binary classification, particularly with linearly separable data. KNN is a non-parametric, distance-based model effective for small datasets, though computationally intensive with larger ones. DT is intuitive and interpretable but requires careful tuning to avoid overfitting. NB, a probabilistic classifier, performs well on high-dimensional data and is particularly effective in text-based applications. In addition to ML models, several DL architectures are also implemented, including DNN, CNN, RNN, LSTM, BiLSTM, and GRU. DNN leverages multiple hidden layers to capture non-linear relationships in large datasets. CNN is designed to detect local patterns, making it suitable for sequential data such as IoT traffic. RNN is tailored for temporal data, learning dependencies across time steps. LSTM addresses the vanishing gradient problem, enabling the retention of long-range dependencies. BiLSTM processes input in both forward and backward directions, enriching contextual understanding. GRU, a streamlined variant of LSTM, offers computational efficiency while preserving the ability to learn long-term patterns. After selecting the models, they need a set of hyperparameters that should be optimized to reach the maximum performance. The Grid Search technique is used to modify internal parameters and perform hyperparameter optimization so they can discover patterns in the data with the purpose of intrusion detection in IoT networks. Grid Search determines the highest performing combination of hyperparameters by methodically testing their combinations. This is essential to the process of optimizing the models' performance and making them

generalize well on unseen data. Although the process is computationally demanding, it highly enhances the performance of the model by optimizing the parameters. Description of the optimized hyperparameters for the ML and DL models is provided in Table. 1 and Table. 2, respectively.

Table. 1: Hyperparameter tuning for ML models

Algorithm	Parameter	Value
RF	n_estimators	100
	max_depth	None
	min_samples_split	2
	min_samples_leaf	1
	max_features	"sqrt"
	bootstrap	TRUE
GB	n_estimators	100
	learning_rate	0.1
	max_depth	3
	min_samples_split	2
	min_samples_leaf	1
	subsample	1
AdaBoost	n_estimators	50
	learning_rate	1
	base_estimator	DecisionTreeClassifier(max_depth=1)
LR	penalty	"l2"
	C	1
	solver	"lbfgs"
	max_iter	100
KNN	n_neighbors	5
	metric	"minkowski"
	p	2 (Euclidean)
	weights	"uniform"
DT	criterion	"gini"
	max_depth	None
	min_samples_split	2
	min_samples_leaf	1
	max_features	None
NB	var_smoothing	1.00E-09
SVM	C	1
	kernel	"rbf"
	gamma	"scale"
	degree	3
K-Means	n_clusters	8
	init	"k-means++"
	max_iter	300
	n_init	10
	tol	1.00E-04

3.3 Model Evaluation

The trained optimized models are evaluated with performance assessed using a comprehensive suite of metrics, including accuracy, precision, recall, F1-score, AUC-ROC, Matthews Correlation Coefficient (MCC), Log Loss, and Hamming Loss. Additionally, confusion matrix components, True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN), are analyzed to provide granular insight into classification behavior. This multi-metric evaluation enables a detailed performance comparison across models, supporting informed decision-making and guiding the selection of the most effective model for deployment within MQTT-based IoT environments.

3.4 Experimental Setup

The experiments were carried out using a blend of hardware and software resources. Model training and evaluation were executed on Google Colab with GPU acceleration, ensuring efficient computation, and supplemented by a local laptop equipped with an Intel(R) Core (TM) Ultra 7 155U processor (2.10 GHz) and 16 GB RAM. For data handling and numerical operations, Pandas and NumPy were employed. Deep learning applications were developed using TensorFlow along with its high level API Keras, which facilitated the construction and deployment of complex architectures. Additionally, Scikit-learn provided a comprehensive suite of tools for preprocessing, model fitting, hyperparameter optimization, performance assessment, and other utilities essential to building and validating the proposed framework.

4. Results Analysis

This section presents the results obtained from the implementation of the models, with a focus on a side by side comparison and analysis of their performance. The evaluation is based on each model's competence in classifying instances accurately, minimizing prediction errors, and producing well calibrated probability estimates. This analysis provides insight into the relative strengths and limitations of the models to handle the unique characteristics of MQTT traffic and their robustness against varied attack patterns in IoT networks.

4.1 True Positive

True Positive (TP) refers to the number of actual attack instances identified correctly.

Table. 2: DL key layers with 0.001 learning rate and 0.5 dropout

Model	Input Shape	Units	Filters/ Kernels	Layers	Batch Size	Epochs	Activation	Sequence Length
DNN	(100,)	128	N/A	3	64	50	ReLU +Softmax	N/A
CNN	(64, 64, 3)	N/A	64 filters, 3×3 kernel	5	64	50	ReLU +Softmax	N/A
RNN	(100, 50)	128	N/A	2	64	30	Softmax	100
LSTM	(100, 50)	128	N/A	2	64	30	Softmax	100
BiLSTM	(100, 50)	128	N/A	2	64	30	Softmax	100
GRU	(100, 50)	128	N/A	2	64	30	Softmax	100

In intrusion detection systems, especially within MQTT-based IoT networks, a high TP count indicates strong detection capability and minimal oversight of malicious traffic. While TP alone does not account for false alarms or missed detections, it remains a critical indicator of a model's effectiveness in recognizing threats. The TP results are presented in Fig. 2.

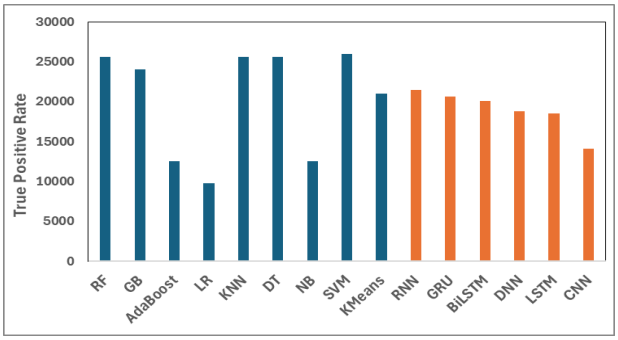

Fig. 2: True positive comparison

The TP results reveal clear distinctions in detection effectiveness between the models within the proposed framework. Among ML models, SVM achieved the highest TP count at 26,000, followed closely by KNN (25,588), RF (25,586), and DT (25,586). These models consistently demonstrated strong detection capabilities, identifying nearly all malicious instances. GB also performed well with 24,060 TPs, while AdaBoost (12,528), NB (12,526), and LR (9,722) showed notably lower TP rates, indicating reduced sensitivity to complex attack patterns. In comparison, DL models exhibited more moderate TP performance. RNN led the group with 21,430 TPs, followed by GRU (20,654) and BiLSTM (20,050), suggesting that sequence-aware architectures are effective in capturing temporal attack behaviors. DNN (18,750), LSTM (18,540), and CNN (14,110) recorded lower TP counts, indicating that these models may be less suited to the specific characteristics of MQTT traffic. Notably, KMeans, an unsupervised ML model, achieved 21,000 TPs, comparable to the top DL models, highlighting its potential in scenarios with limited labeled data. Overall, ML models outperformed DL models in TP detection, with SVM, KNN, RF, and DT forming a clear top tier. This suggests that for MQTT-based IoT networks, traditional ML classifiers currently offer superior accuracy in identifying threats based on TP outcomes alone.

4.2 True Negative

True Negative (TN) refers to the number of benign or normal traffic instances correctly identified as non-malicious. In intrusion detection systems, a high TN count indicates that the model effectively avoids false alarms, preserving system stability and reducing unnecessary interventions. Evaluating TN alongside TP provides a more complete picture of a model's reliability in distinguishing between legitimate and harmful activity. TN results in Fig. 3 refer to the number of cases that are correctly predicted as negative by each model.

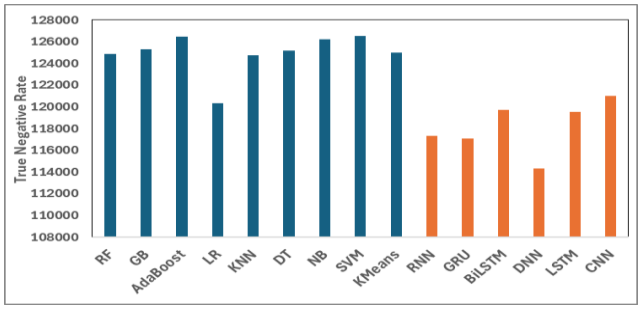

Fig. 3: True negative comparison

The TN results show that several machine learning models excel at correctly identifying normal traffic. SVM achieved the highest TN count at 126,500, followed closely by AdaBoost (126,472) and NB (126,241), indicating strong performance in minimizing false positives. GB (125,305), DT (125,188),

and KMeans (125,000) also performed well, suggesting consistent reliability across ensemble and clustering-based approaches. RF and KNN followed with 124,897 and 124,773 TNs, respectively, maintaining solid but slightly lower performance. LR recorded a lower TN count of 120,313, reflecting reduced precision in distinguishing benign traffic. Among deep learning models, BiLSTM (119,740), LSTM (119,510), and CNN (121,000) showed moderate TN performance, while RNN (117,320), GRU (117,050), and DNN (114,300) had the lowest TN rates in the group. These results suggest that while DL models are effective in capturing temporal patterns, they may be more prone to misclassifying normal traffic compared to ML models. Overall, ML models, particularly SVM, AdaBoost, and NB, demonstrated superior TN performance, reinforcing their strength in maintaining low false positive rates. This complements their high TP results and supports their suitability for reliable intrusion detection in MQTT based IoT networks. DL models, while valuable for sequential pattern recognition, showed more variability in TN outcomes, indicating room for improvement in distinguishing benign traffic with greater precision

4.3 False Positive

False Positive (FP) refers to the number of benign or normal traffic instances incorrectly classified as malicious. A high FP rate can lead to unnecessary alerts, wasted resources, and reduced trust in the system. Therefore, minimizing FP is essential for maintaining operational efficiency and ensuring that legitimate traffic is not disrupted. The FP results are provided in Fig. 4 for each model.

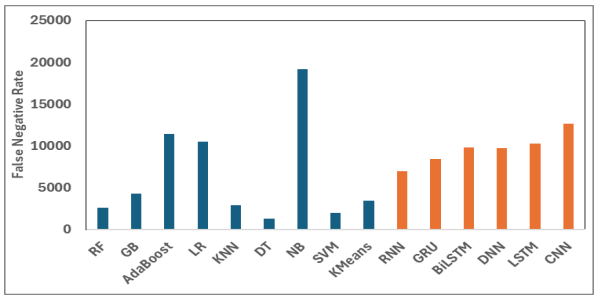

Fig. 4: False positive comparison

The FP results reveal notable differences in how accurately each model distinguishes between malicious and benign traffic. Among machine learning models, DT achieved the lowest FP count (354), indicating exceptional precision in identifying normal traffic. SVM (3,000), KNN (3,212), and RF (4,141) also performed well, maintaining low FP rates and reinforcing their reliability in minimizing false alarms. GB and KMeans followed with moderate FP counts of 5,975 and 5,000, respectively. In contrast, AdaBoost (17,049), LR (16,269), and NB (19,417) recorded significantly higher FP rates, suggesting a tendency to overclassify benign traffic as malicious. These elevated FP values may reflect limitations in model generalization or sensitivity to feature noise. Among deep learning models, FP rates were generally higher than those of the top-performing ML models. RNN (8,600), GRU (10,032), BiLSTM (9,700), DNN (10,300), LSTM (9,000), and CNN (8,800) all showed moderate to high FP counts. While these models are effective in capturing temporal patterns, their elevated FP rates suggest challenges in distinguishing subtle differences between normal and attack traffic in MQTT-based environments. Overall, the analysis highlights that ML models, particularly DT, SVM, KNN, and RF, are more effective in minimizing false positives compared to their DL counterparts. This reinforces their suitability for deployment in MQTT-based IoT networks, where maintaining low FP rates is critical for preserving system integrity and user trust.

4.4 False Negatives

False Negative (FN) refers to the number of actual attack instances that a model fails to detect and mistakenly classifies as benign. A high FN rate is particularly concerning, as it allows malicious traffic to pass through undetected, potentially compromising the network. Therefore, minimizing FN is essential for ensuring comprehensive threat coverage and maintaining the integrity of MQTT-based IoT networks. FN results of the models are compared in Fig. 5. The FN results reveal significant variation in detection reliability across both machine learning and deep learning models. Among machine learning models, DT achieved the lowest FN count (1,316), followed closely by SVM (2,000), RF (2,631), and KNN (2,938). These models demonstrated strong sensitivity to malicious traffic, missing very few attack instances. GB also performed reasonably well with 4,286 FNs, maintaining a balance between detection and precision.

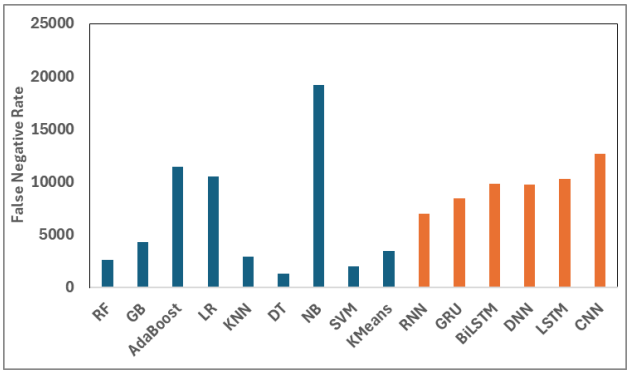

Fig. 5: False negative comparison

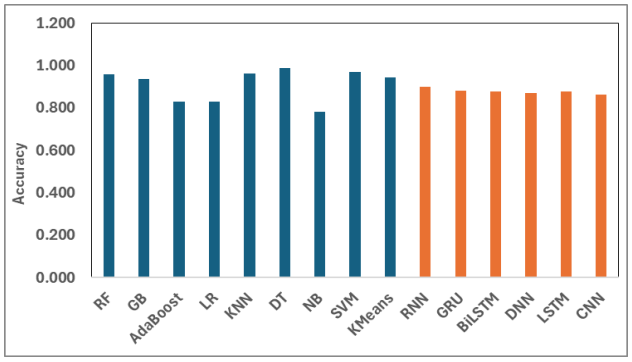

Fig. 6: Accuracy comparison

In contrast, AdaBoost (11,445), LR (10,494), and NB (19,155) recorded substantially higher FN rates, indicating reduced effectiveness in identifying threats. NB, in particular, showed the weakest performance, missing nearly 20,000 attack instances, which raises concerns about its suitability for high-stakes intrusion detection. Among deep learning models, RNN (7,000), GRU (8,470), and BiLSTM (9,850) formed a mid-tier group, with moderate FN rates that suggest reasonable but not optimal sensitivity. DNN (9,750), LSTM (10,280), and CNN (12,646) exhibited higher FN counts, indicating a tendency to overlook a significant portion of malicious traffic. These results suggest that while DL models can capture complex patterns, they may struggle to consistently detect all attack instances in MQTT-based environments. Together, the analysis highlights that ML models, particularly DT, SVM, RF, and KNN, outperform DL models in minimizing false negatives. This reinforces their role as reliable components in the proposed framework, ensuring that threats are accurately identified and mitigated with minimal oversight.

4.5 Accuracy

Accuracy measures the proportion of correctly classified instances, both malicious and benign, out of the total number of samples, which provides a general indication of a model's overall performance. While accuracy alone does not reveal the balance between false positives and false negatives, it remains a useful metric for comparing the broad effectiveness of different models. The overall accuracy of each model in the framework is provided in Fig. 6. The accuracy results reveal that machine learning models generally outperform deep learning models in overall classification performance.

DT achieved the highest accuracy at 0.9890, followed closely by SVM (0.9682), KNN (0.9607), and RF (0.9569). These models consistently demonstrated strong performance across both malicious and benign traffic, confirming their reliability in MQTT-based IoT environments. GB also maintained a high accuracy level at 0.9357, while KMeans followed with 0.9449, showing that even unsupervised approaches can yield competitive results. In contrast, AdaBoost (0.8299), LR (0.8293), and NB (0.7825) recorded lower accuracy scores, indicating reduced effectiveness in handling the complexity of intrusion patterns. Among deep learning models, RNN (0.8989), GRU (0.8815), BiLSTM (0.8773), LSTM (0.8775), DNN (0.8690), and CNN (0.8630) all fell below the top performing ML models. While these DL models offer advantages in modeling sequential data, their overall accuracy suggests they may struggle with generalizing across diverse traffic types in MQTT based networks. Overall, the analysis confirms that ML models, particularly DT, SVM, KNN, and RF, achieve superior accuracy in intrusion detection tasks. This supports their selection within the proposed framework, ensuring high reliability and broad coverage in identifying both normal and malicious traffic

4.6 Precision

Precision measures the proportion of correctly identified malicious instances out of all instances classified as malicious. A high precision score indicates that a model generates fewer false alarms, making it particularly valuable where minimizing unnecessary alerts is critical for operational efficiency. Precision results in Fig. 7 compare how well a model can predict correctly positive instances so that the model maintains low false positives.

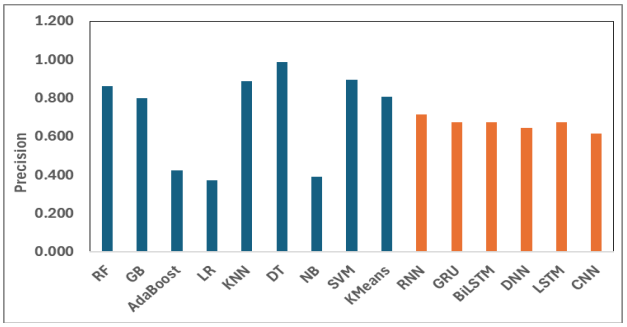

Fig. 7: Precision comparison

The precision results reveal a clear performance advantage for several machine learning models. DT achieved the highest precision score at 0.9863, indicating exceptional accuracy in identifying true threats with minimal false positives. SVM (0.8966), KNN (0.8885), and RF (0.8607) also performed strongly, confirming their reliability in producing accurate threat classifications. GB and KMeans followed with respectable scores of 0.8011 and 0.8077, respectively, maintaining solid precision across diverse traffic patterns. In contrast, AdaBoost (0.4236), LR (0.3741), and NB (0.3921) recorded substantially lower precision values, suggesting a tendency to misclassify benign traffic as malicious. These models may be more sensitive to noise or less effective in distinguishing subtle attack features. Among deep learning models, RNN achieved the highest precision at 0.7136, followed by GRU (0.6731), BiLSTM (0.6740), and LSTM (0.6732), forming a mid-tier group. DNN (0.6454) and CNN (0.6159) showed the lowest precision among DL models, indicating a higher rate of false positives and reduced reliability in threat identification. Overall, the analysis highlights that ML models, particularly DT, SVM, KNN, and RF, outperform DL models in precision, making them more suitable for environments where accurate threat classification and low false alarm rates are essential. These findings support the framework's emphasis on precision-optimized models for securing MQTT - based IoT networks.

4.7 Recall

Recall measures a model's ability to correctly identify actual malicious instances out of all existing attack samples. A high recall score indicates strong sensitivity to threats, minimizing the number of undetected attacks (false negatives). In intrusion detection systems, especially within MQTT-based IoT networks, high recall is essential for ensuring comprehensive threat coverage. Recall results of the models within the framework are compared in Fig. 8.

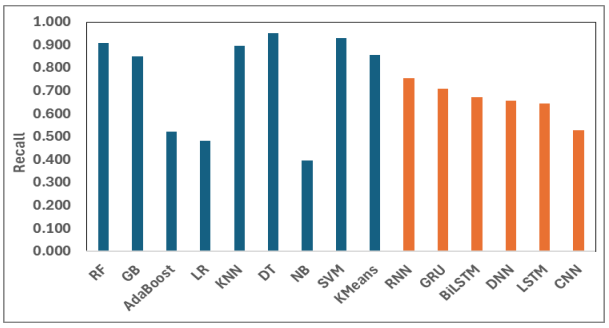

Fig. 8: Recall comparison

The recall results reveal that several machine learning models excel in identifying malicious traffic. DT achieved the highest recall score at 0.9511, followed by SVM (0.9286), RF (0.9068), and KNN (0.8970). These models consistently demonstrated strong sensitivity, successfully detecting the majority of attack instances and minimizing false negatives. GB and KMeans also performed well, with recall scores of 0.8488 and 0.8571, respectively, indicating reliable detection capabilities. In contrast, AdaBoost (0.5226), LR (0.4809), and NB (0.3954) recorded significantly lower recall scores, suggesting limited effectiveness in capturing malicious activity. These models may struggle with complex or subtle attack patterns, leading to higher rates of undetected threats. Among deep learning models, RNN (0.7538) led the group, followed by GRU (0.7092), BiLSTM (0.6706), DNN (0.6579), LSTM (0.6433), and CNN (0.5274). While these models offer valuable temporal modeling capabilities, their recall scores were generally lower than those of the top performing ML models, indicating a greater tendency to miss attack instances. The analysis shows that ML models, particularly DT, SVM, RF, and KNN, outperform DL models in recall, making them more effective for comprehensive threat detection in MQTT based IoT networks. These findings reinforce the importance of selecting models with high sensitivity to ensure robust security coverage.

4.8 F1-score

The F1-score is the harmonic mean of precision and recall, offering a balanced measure of a model's ability to correctly identify malicious traffic while

minimizing false positives and false negatives. It is particularly useful in tasks where both types of errors can have serious consequences. A higher F1-score indicates better overall classification performance. F1-score results are compared in Fig. 9.

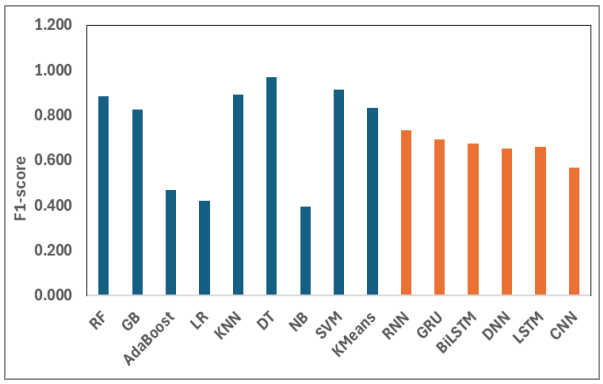

Fig. 9: F1-score comparison

The F1-score results show that several machine learning models deliver strong and balanced performance. DT achieved the highest F1-score at 0.9684, reflecting its exceptional precision and recall in identifying threats. SVM (0.9123), KNN (0.8927), and RF (0.8831) also performed very well, confirming their reliability in maintaining both detection sensitivity and classification accuracy. GB and KMeans followed with respectable scores of 0.8242 and 0.8317, indicating consistent performance across varied traffic patterns. In contrast, AdaBoost (0.4679), LR (0.4208), and NB (0.3938) recorded substantially lower F1-scores, suggesting imbalanced performance and reduced effectiveness in distinguishing between malicious and benign traffic. Among deep learning models, RNN achieved the highest F1-score at 0.7332, followed by GRU (0.6907), BiLSTM (0.6723), LSTM (0.6579), DNN (0.6516), and CNN (0.5682). While these models offer valuable temporal modeling capabilities, their F1-scores were generally lower than those of the top-performing ML models, indicating less consistent performance in balancing precision and recall. Therefore, from the results, it is concluded that ML models, particularly DT, SVM, KNN, and RF, outperform DL models in terms of F1-score, making them more suitable for robust and balanced intrusion detection in MQTT-based IoT networks. These findings reinforce the framework's emphasis on selecting models that deliver high overall classification quality.

4.9 ROC AUC

The Receiver Operating Characteristic Area Under Curve (ROC AUC) metric evaluates a model's ability to distinguish between malicious and benign traffic. A higher ROC AUC score indicates better overall classification performance across all thresholds, reflecting both sensitivity and specificity. It is particularly valuable in intrusion detection, where balancing detection and false alarm rates is critical. ROC AUC results in Fig. 10 compare the model's performance.

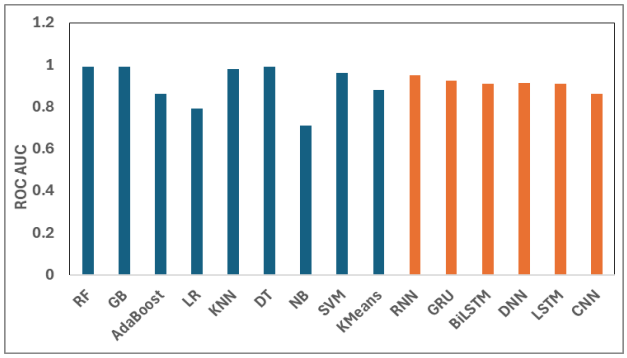

Fig. 10: ROC AUC comparison

The above results show that several machine learning models achieved near-perfect discrimination capability. RF, GB, and DT each reached a score of 0.99, indicating exceptional performance in distinguishing between attack and normal traffic across varying decision thresholds. KNN also performed strongly with a score of 0.98, while SVM followed with 0.96, confirming its robust classification ability. KMeans, despite being unsupervised, achieved a respectable ROC AUC of 0.88, outperforming several supervised models and demonstrating its potential in scenarios with limited labeled data. AdaBoost (0.86) and LR (0.79) showed moderate performance, while NB recorded the lowest ROC AUC among ML models at 0.71, suggesting limited reliability in class separation. Among deep learning models, RNN led with a score of 0.9512, followed by GRU (0.9231), DNN (0.9115), BiLSTM (0.9103), and LSTM (0.909). These models demonstrated a strong capacity to capture complex patterns in sequential data. CNN, while effective in spatial feature extraction, recorded the lowest ROC AUC among DL models at 0.8601, indicating reduced effectiveness in distinguishing MQTT traffic classes. Therefore, the analysis concludes that ML models, particularly RF, GB, DT, and KNN,

outperform DL models in ROC AUC, offering superior class discrimination for intrusion detection in MQTT-based IoT networks. These findings reinforce the framework's prioritization of models that deliver high classification reliability across diverse operating conditions.

4.10 MCC

Matthews Correlation Coefficient (MCC) is a balanced metric that evaluates classification performance by considering true and false positives and negatives. Unlike accuracy, MCC remains reliable even when class distributions are imbalanced, making it particularly valuable for intrusion detection tasks. An MCC score closer to 1 indicates strong predictive performance, while values near 0 suggest poor or random classification. MCC results in Fig. 11 are particularly useful to provide a general performance evaluation of the models within the detection framework.

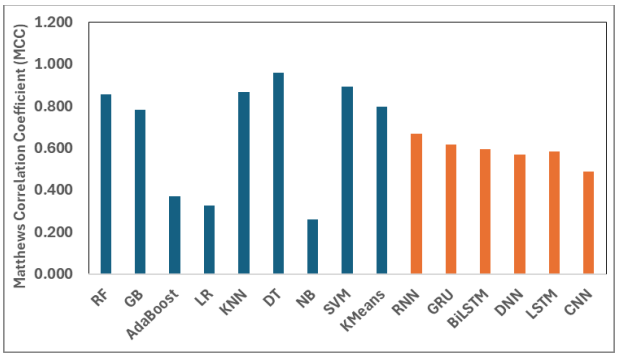

Fig. 11: MCC comparison

The MCC results reveal that machine learning models generally outperform deep learning models in balanced classification performance. DT achieved the highest MCC score at 0.9620, reflecting near-perfect agreement between predicted and actual classes. SVM (0.8931), KNN (0.8687), and RF (0.8572) also demonstrated excellent performance, confirming their robustness in handling both attack and benign traffic with minimal misclassification. GB (0.7854) and KMeans (0.7993) followed with solid scores, indicating consistent and reliable behavior across varied traffic patterns. In contrast, AdaBoost (0.3708), LR (0.3260), and NB (0.2612) recorded substantially lower MCC values, suggesting weaker overall classification consistency and higher susceptibility to errors. Among deep learning models, RNN led with an MCC of 0.6713, followed by GRU (0.6178), BiLSTM (0.5968), LSTM (0.5835), DNN (0.5710), and CNN (0.4894). While these models offer valuable temporal modeling capabilities, their MCC scores indicate less balanced performance compared to top ML models, with a greater tendency toward misclassification in either direction. Overall, the analysis of the results confirms that ML models, especially DT, SVM, KNN, and RF, achieve superior MCC scores, making them more reliable for intrusion detection in MQTT-based IoT networks. These findings reinforce the framework's emphasis on selecting models that deliver consistent and balanced classification outcomes.

4.11 Log & Hamming loss

Hamming loss and log loss both measure the deviation of a model's predictions from true labels but in distinct ways. Hamming loss quantifies the prediction error rate by calculating the fraction of incorrect predictions relative to the total number of predictions, with lower values indicating fewer misclassifications and higher accuracy. Conversely, log loss assesses the uncertainty of a model's predictions by penalizing confident yet incorrect classifications, which is crucial in security contexts for identifying models that balance accurate classification with appropriate uncertainty for risk sensitive decisions. Lower log loss values reflect better-calibrated probability estimates and more reliable predictions. Together, these metrics provide complementary insights where hamming loss indicates the frequency of errors, while log loss evaluates the severity of errors when the model is confident. The comparison of the log and hamming loss values for each model in the detection framework is provided in Fig. 12.

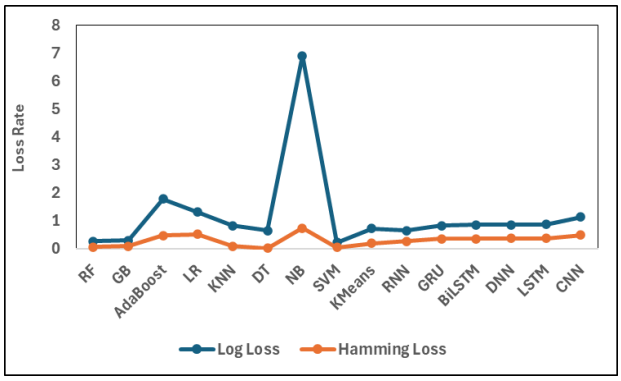

Fig. 12: Log loss and hamming loss comparison

The results show that several machine learning models achieved low error rates across both metrics. SVM recorded the lowest Log Loss (0.23) and a very low Hamming Loss (0.05), indicating highly confident and accurate predictions. RF (0.27, 0.07) and GB (0.30, 0.09) also performed well, maintaining strong calibration and low misclassification rates. DT stood out with the lowest Hamming Loss (0.02) and a competitive Log Loss (0.65), confirming its precision and reliability. KNN (0.83, 0.09) and KMeans (0.72, 0.20) showed moderate performance, with slightly higher uncertainty and misclassification rates but still within acceptable bounds. In contrast, AdaBoost (1.78, 0.48), LR (1.32, 0.52), and NB (6.90, 0.74) recorded significantly higher losses, suggesting poor confidence calibration and frequent misclassifications. NB, in particular, showed the weakest performance across both metrics. Among deep learning models, RNN achieved the lowest Log Loss (0.658) and Hamming Loss (0.275) within its group, followed by GRU (0.826, 0.355), BiLSTM (0.865, 0.366), DNN (0.8652, 0.3715), LSTM (0.88, 0.377), and CNN (1.142, 0.495). These results indicate that while DL models offer valuable pattern recognition capabilities, they generally exhibit higher uncertainty and misclassification rates compared to top-performing ML models. Overall, the analysis confirms that ML models, particularly SVM, DT, RF, and GB, outperform DL models in both log loss and hamming loss, offering more confident and accurate predictions for intrusion detection in MQTT-based IoT networks. These findings reinforce the framework's prioritization of models that minimize prediction error and enhance classification reliability.

5. Conclusion

This work introduces an AI-based intrusion detection framework that integrates both machine learning and deep learning models within MQTT-based IoT environments. Through extensive experimentation on real world traffic data, the models are evaluated for their ability to accurately distinguish between benign and malicious activity, minimize classification errors, and maintain consistent decision-making across varied attack scenarios. The comparative analysis revealed that machine learning models generally outperformed deep learning counterparts in terms of detection reliability, computational efficiency, and overall robustness. Among them, certain tree-based algorithms exhibited exceptional performance, offering a strong balance between precision and generalization. Deep learning models, while moderately effective in capturing sequential patterns, showed limitations in consistency and error control, particularly under complex or imbalanced conditions. These findings underscore the practical advantages of lightweight, interpretable machine learning models for scalable and dependable intrusion detection in resource-constrained IoT systems. The study provides a foundation for future work exploring hybrid or ensemble approaches that combine the strengths of both paradigms to further enhance security in MQTT-based networks. Looking forward, this work establishes a basis for exploring hybrid or ensemble approaches that integrate ML and DL strengths. Reinforcement learning (RL) can also be investigated as a complementary direction to enable adaptive intrusion detection capable of responding dynamically to evolving attack behaviors in MQTT-based networks.

Funding

This research received no external funding

Data Availability Statement

Data sharing is not applicable to this article as no datasets were generated or analyzed.

Conflict of Interest

The authors declared "No Conflict of Interest".

CRediT authorship contribution statement

Both authors contributed to this manuscript equally. Both authors have read and agreed to the published version of the manuscript.

References

[1] R. Chataut, A. Phoummalayvane, R. Akl "Unleashing the power of IoT: A comprehensive review of IoT applications and future prospects in healthcare, agriculture, smart homes, smart cities, and Industry 4.0", *Sensors*, Vol. 23, No. 16, art. no. 7194, 2023. https://doi.org/10.3390/s23167194

[2] T. Xinyu, K. Pekka, H. Ismo "A design and modeling approach for resource constrained internet of things devices", *Design Automation for Embedded Systems*, Vol. 29, No. 4, 2025. https://doi.org/10.1007/s10617-025-09294-6

[3] A. J. Hintaw, S. Manickam, S. Karuppayah, M. F. Aboalmaaly "A brief review on MQTT's security issues within the Internet of Things (IoT)", *Journal of Communications*, Vol. 14, No. 6, pp. 463–469, 2019. https://doi.org/10.12720/jcm.14.6.463-469

[4] T. Mazhar, D. B. Talpur, T. Al Shloul, Y. Y. Ghadi, I. Haq, I. Ullah, K. Ouahada, H. Hamam "Analysis of IoT security challenges and its solutions using artificial intelligence", *Brain Sciences*, Vol. 13, No. 4, art. no. 683, 2023. https://doi.org/10.3390/brainsci13040683

[5] A. Kumar, J. A. Gutierrez "Impact of machine learning on intrusion detection systems for the protection of critical infrastructure", *Information*, Vol. 16, No. 7, art. no. 515, 2025. https://doi.org/10.3390/info16070515

[6] F. Alwahedi, A. Aldhaheri, M. A. Ferrag, A. Battah, N. Tihanyi "Machine learning techniques for IoT security: Current research and future vision with generative AI and large language models", *Internet of Things and Cyber-Physical Systems*, Vol. 4, pp. 167–185, 2024. https://doi.org/10.1016/j.iotcps.2023.12.003

[7] M. S. Ahsan, A. S. K. Pathan "A comprehensive survey on the requirements, applications, and future challenges for access control models in IoT: The state of the art", *IoT*, Vol. 6, No. 1, art. no. 9, 2025. https://doi.org/10.3390/iot6010009

[8] Ch. Amarendra, D. Balakotaiah, P. V. Rajulu, A. P. Sridhar, T. K. Mohana "Decoding the Internet of Things: A comprehensive survey paper", *International Transactions on Electrical Engineering and Computer Science*, Vol. 4, No. 2, pp. 103–118, 2025. https://doi.org/10.62760/iteecs.4.2.2025.136

[9] J. D. Gadze, A. A. Bamfo-Asante, J. O. Agyemang, H. Nunoo-Mensah, K. A. B. Opare "An investigation into the application of deep learning in the detection and mitigation of DDoS attack on SDN controllers", *Technologies*, Vol. 9, No. 1, art. no. 14, 2021. https://doi.org/10.3390/technologies9010014

[10] N. Ahuja, G. Singal, D. Mukhopadhyay, N. Kumar "Automated DDoS attack detection in software defined networking", *Journal of Network and Computer Applications*, Vol. 187, art. no. 103108, 2021. https://doi.org/10.1016/j.jnca.2021.103108

[11] Z. Wang, Y. Zeng, Y. Liu, D. Li "Deep belief network integrating improved kernel-based extreme learning machine for network intrusion detection", *IEEE Access*, Vol. 9, pp. 16062–16091, 2021. https://doi.org/10.1109/ACCESS.2021.3051074

[12] S. Dotcenko, A. Vladyko, I. Letenko "A fuzzy logic based information security management for software defined networks", *16th International Conference on Advanced Communication Technology (ICACT)*, pp. 167–171, 2014. https://doi.org/10.1109/ICACT.2014.6778942

[13] T. Abhiroop, S. Babu, B. S. Manoj "A machine learning approach for detecting DoS attacks in SDN switches", *2018 National Conference on Communications (NCC)*, pp. 1-6, 2018. https://doi.org/10.1109/NCC.2018.8600196

[14] K. S. Sahoo, A. Iqbal, P. Maiti, B. Sahoo "A machine learning approach for predicting DDoS traffic in software defined networks", *2018 International Conference on Information Technology (ICIT)*, pp. 199–203, 2018. https://doi.org/10.1109/ICIT.2018.00049

[15] M. Carlier, K. Steenhaut, A. Braeken "Symmetric key based security for multicast communication in wireless sensor networks", *Computers*, Vol. 8, No. 1, art. no. 27, 2019. https://doi.org/10.3390/computers8010027

[16] P. Shabisha, A. Braeken, P. Kumar, K. Steenhaut "Fog-orchestrated and server-controlled anonymous group authentication and key agreement", *IEEE Access*, Vol. 7, pp. 150247–150261, 2019. https://doi.org/10.1109/ACCESS.2019.2946713

[17] A. Chandramouli, A. Choudhury, A. Patra "A survey on perfectly secure verifiable secret-sharing", *ACM Computing Surveys*, Vol. 54, No. 11s, pp. 1-36, 2022. https://doi.org/10.1145/3512344

[18] M. Tiloca, K. Nikitin, S. Raza "Axiom: DTLS-based secure IoT group communication", *ACM*

Transactions on Embedded Computing Systems, Vol. 16, No. 3, pp. 1-29, 2017. https://doi.org/10.1145/3047413

[19] H. Zeghida, M. Boulaiche, R. Chikh, A. M. Bamhdi, A. L. B. Barros, D. Zeghida, A. Patel "Enhancing IoT cyber attacks intrusion detection through GAN-based data augmentation and hybrid deep learning models for MQTT network protocol cyber attacks", *Cluster Computing*, Vol. 28, art. no. 58, 2025. https://doi.org/10.1007/s10586-024-04752-5

[20] I. H. Putro, T. Ahmad, R. M. Ijtihadie "Enhancing MQTT intrusion detection in IoT using machine learning and feature engineering", *IEEE Open Journal of the Communications Society*, Vol. 6, pp. 7855–7884, 2025. https://doi.org/10.1109/OJCOMS.2025.3610132

[21] M. Solanki, S. Gupta "A novel intrusion detection framework using ensemble learning in MQTT IoT applications", *Annals of Mathematics and Artificial Intelligence*, 2025. https://doi.org/10.1007/s10472-025-09993-7

[22] M. Swain, N. Tripathi, K. Sethi "Identifying communication sequence anomalies to detect DoS attacks against MQTT", *Computers & Security*, Vol. 157, art. no. 104526, 2025. https://doi.org/10.1016/j.cose.2025.104526

[23] H. Run-Ze, S. Jun-Jian, Q. Su-Juan, J. Zheng-Ping, and F. Gao "QGAN-based data augmentation for hybrid quantum–classical neural networks", *Chinese Journal of Physics*, Vol. 97, pp. 1453-1463, 2025. https://doi.org/10.1016/j.cjph.2025.07.017

[24] A. Aqachtoul, K. Karam, A. Elamrani, M. Najib, N. Rafalia, M. Bakhouya "MQTTEEB-D: A real-world IoT cybersecurity dataset for AI-powered threat detection in MQTT networks", *Data in Brief*, Vol. 62, art. no. 111897, 2025. https://doi.org/10.1016/j.dib.2025.111897

Copyright: © 2026 by the authors, Licensee ITEECS, India. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

***