Date of publication SEP-30, 2025, date of current version MAY-29, 2025.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.24.3.4198

Transforming IIoT Security Leveraging Deep Learning and Feature Selection for Superior Intrusion Detection

LAHCEN IDOUGLID, SAID TKATEK, KHALID ELFAYQ
Computer Sciences Research Laboratory, Ibn Tofail University, Kenitra, Morocco

Corresponding author: Lahcen Idouglid (e-mail: lahcen.idouglid@uit.ac.ma).

ABSTRACT The Industrial Internet of Things (IIoT) has revolutionized industrial operations but has also brought forth significant cybersecurity challenges, demanding the development of advanced Intrusion Detection Systems (IDS). This study presents a feature-driven approach to enhance IDS performance in IIoT environments. By utilizing Recursive Feature Elimination (RFE) combined with Mutual Information (MI) for feature selection, we identified the most relevant attributes from the UNSW-NB15 dataset, improving detection accuracy while reducing computational complexity. Several deep learning models, including Convolutional Neural Networks (CNN), Residual Neural Networks (ResNet), Long Short-Term Memory (LSTM), and Bidirectional LSTM (BiLSTM), were evaluated. Among them, BiLSTM delivered the best performance, achieving a recall of 96.96%, an F1-score of 97.06%, and a Matthews Correlation Coefficient (MCC) of 0.93, outperforming other models in detecting complex attack patterns. However, its high computational cost, with training time exceeding 3500 seconds, underscores the need for optimization for real-time deployment. The results highlight the potential of combining feature selection techniques with deep learning models to enhance IDS for IIoT. Future work will focus on optimizing BiLSTM for faster deployment, integrating hybrid models, and testing across diverse datasets to further improve real-time security solutions for IIoT environments.

KEYWORDS IIoT Security; Intrusion Detection Systems; Deep Learning; Feature Selection; BiLSTM; Recursive Feature Elimination

I. INTRODUCTION

The IIoT represents a transformative shift in industrial processes, enabling unprecedented levels of automation, efficiency, and real-time data analytics. By interconnecting devices, sensors, and systems, IIoT facilitates innovations like predictive maintenance, process optimization, and intelligent decision-making. These advancements promise significant economic benefits across industries such as manufacturing, energy, healthcare, and transportation. However, the proliferation of IIoT devices introduces considerable cybersecurity challenges, given their distributed and interconnected nature. Attackers can exploit vulnerabilities in IIoT networks to compromise critical infrastructure, leading to devastating financial, operational, and even human consequences. Addressing these security challenges is critical to unlocking the full potential of IIoT systems [1], [2].

IDS are essential components of any cybersecurity strategy, designed to monitor network traffic and identify anomalous or malicious activities. Traditional IDS methods, based on signature-based detection, offer limited efficacy against novel or unknown attacks. With the rising complexity and volume of IIoT data, traditional techniques often fall short in terms of scalability and accuracy. As a result, machine learning (ML) and deep learning (DL) approaches have gained traction for their ability to learn intricate patterns in high-dimensional data and detect sophisticated attack vectors. Recent studies highlight the superior performance of deep learning models, such as CNNs, LSTM networks, and BiLSTM architectures, in securing IIoT environments [3], [4], [5].

A. FEATURE SELECTION AND DATA PREPROCESSING

Feature selection is a critical step in building robust IDS, particularly when working with high-dimensional datasets such as UNSW-NB15 or NSL-KDD. These datasets contain diverse features, some of which may be redundant or irrelevant, leading to increased computational costs and reduced detection accuracy. RFE with MI has emerged as a powerful technique for identifying the most relevant features, enabling models to focus on the most impactful attributes. Studies show that effective feature selection not only enhances model performance but also reduces the training time, making the approach suitable for resource-constrained IIoT environments [6], [7].

Data preprocessing is equally crucial in ensuring the quality and reliability of an IDS. This involves handling missing values, scaling numerical features, and transforming skewed distributions. Techniques like outlier detection and log transformation can mitigate the impact of extreme values, improving model robustness. Properly processed data serves as a strong foundation for training machine learning and deep learning models [8], [9].

B. DEEP LEARNING MODELS FOR IIOT SECURITY

Deep learning architectures has shown remarkable success in improving IDS performance due to their ability to capture complex patterns and adapt to evolving threats. Commonly used models in IDS include:

CNN: Effective for detecting spatial patterns in data, CNNs are widely used for intrusion detection tasks involving packet-level analysis [6].

LSTM and BiLSTM: Capable of learning temporal dependencies, these models excel in identifying sequential patterns in network traffic. BiLSTM, in particular, enhances detection by processing data in both forward and backward directions, improving accuracy [7].

ResNet: By employing skip connections, ResNet mitigates the vanishing gradient problem, enabling the training of deeper networks for intrusion detection [9].

Hybrid models, combining the strengths of multiple architectures, are gaining popularity for IIoT IDS. For instance, CNN-LSTM models leverage CNN's spatial pattern recognition capabilities and LSTM's sequential analysis strengths, achieving superior performance [6].

Evaluation Metrics and Experimental Validation

Evaluating IDS performance requires a comprehensive approach, encompassing multiple metrics such as accuracy, recall, precision, F1-score, ROC-AUC, and computational efficiency. While accuracy provides an overall assessment, recall is critical for identifying attacks, and F1-score balances precision and recall. Computational costs, such as training and inference times, are equally important for real-time applications in IIoT environments [10], [11], [12].

Stratified K-Fold cross-validation is widely employed to ensure robust model evaluation. This technique maintains the class distribution across training and validation sets, preventing bias in performance estimation. Experiments conducted on benchmark datasets like UNSW-NB15 demonstrate the efficacy of deep learning models when coupled with robust feature selection and preprocessing pipelines [6], [13], [14].

C. REAL-WORLD APPLICABILITY AND CHALLENGES

Despite promising results, deploying IDS in real-world IIoT environments presents challenges. These include the heterogeneity of devices, dynamic network topologies, and the need for lightweight solutions that operate within the constraints of IIoT systems. Addressing these challenges requires continuous advancements in feature selection methods, model optimization techniques, and adaptive learning mechanisms to handle evolving threats [15], [16].

In this study, we propose a feature-driven approach that combines RFE with MI for optimal feature selection and employs state-of-the-art deep learning models, including BiLSTM, CNN, and ResNet, for intrusion detection in IIoT networks. Using the UNSW-NB15 dataset, we conduct a comprehensive evaluation of model performance across multiple metrics, demonstrating the viability of our approach in enhancing IIoT security.

D. CONTRIBUTIONS

In this work, we make several important contributions to improving security in Industrial IoT systems. First, we propose a smart way to select the most useful features from the data by combining two techniques Recursive Feature Elimination and Mutual Information to better handle IIoT network traffic. Then, we compare different deep learning models like CNN, ResNet, LSTM, and BiLSTM using the same dataset and testing setup to ensure a fair comparison. Our results show that BiLSTM performs the best, reaching an impressive F1-score of 0.9706, which makes it a strong choice for detecting threats in complex IIoT environments. Finally, we also look at how long each model takes to train and make predictions, which helps in choosing the right model for real-time applications.

II. RELATED WORK

The IIoT has revolutionized industrial operations by enabling seamless communication and automation across devices. However, this connectivity comes with significant cybersecurity risks, requiring robust IDS. This section explores prior research on IDS methods for IIoT, the role of feature selection in enhancing detection accuracy, and advancements in deep learning techniques for cybersecurity.

A. OVERVIEW OF IDS METHODS FOR IIOT

Traditional IDS techniques, including signature-based and anomaly-based approaches, have been foundational in detecting threats. However, the heterogeneous and real-time nature of IIoT environments often limits their effectiveness in detecting novel attacks [17]. Hybrid IDS frameworks that integrate anomaly-based and signature-based techniques have been proposed to enhance detection capabilities and reduce false positives [15], [18].

Machine learning-based IDS further improved adaptability by analyzing complex traffic patterns. Classical methods like Random Forests (RF), Support Vector Machines (SVM), and Naïve Bayes classifiers have been widely explored [19], but they often depend on handcrafted features, limiting their ability to generalize across different datasets and attack scenarios [20].

B. EXISTING STUDIES ON FEATURE SELECTION TECHNIQUES AND THEIR IMPACT

Feature selection is essential in handling the high dimensionality of IIoT datasets, improving computational efficiency and model performance. RFE combined with MI has emerged as a prominent approach, allowing models to focus on the most relevant features while maintaining interpretability [21], [22], [23]. Studies have shown that RFE with MI enhances both deep learning and traditional models by reducing noise and dimensionality [21].

Other techniques, such as Chi-square tests and Principal Component Analysis (PCA), are also used to simplify datasets. However, PCA sacrifices interpretability by transforming data into a new feature space, whereas RFE with MI maintains transparency in the feature selection process [23]. The integration of feature selection with deep learning has proven effective, with methods like CNN and LSTM networks benefitting significantly from optimized input features [24].

C. ADVANCEMENTS IN DEEP LEARNING FOR CYBERSECURITY APPLICATIONS

Deep learning models have shown great promise in addressing IIoT security challenges due to their capability to learn complex patterns from data. Convolutional Neural Networks (CNNs) have been employed for their efficiency in identifying spatial relationships in network traffic [25], while Recurrent Neural Networks (RNNs) and LSTMmodels have been used to capture temporal dependencies, making them suitable for IIoT's dynamic environments [26], [27].

Bi-directional LSTM (BiLSTM) networks stand out due to their ability to analyze both forward and backward temporal information, improving detection accuracy for sophisticated, multi-stage attacks [28]. ResNet and hybrid architectures like CNN-LSTM have further advanced IDS performance by addressing training challenges such as vanishing gradients and overfitting [29], [30].

Other promising approaches include transfer learning, which leverages pre-trained models for rapid adaptation to new IIoT domains, and ensemble learning, such as bagging and boosting, which combine the strengths of multiple models to improve overall detection robustness [31], [32].

D. SUMMARY OF KEY CONTRIBUTIONS FROM RELATED WORK

The reviewed literature highlights significant progress in IDS for IIoT, particularly with advanced feature selection and deep learning techniques. However, challenges remain, including the need to address data imbalance, optimize computational efficiency, and ensure adaptability to evolving attack vectors. This study builds on these advancements by employing RFE with MI for feature selection, evaluating various deep learning models, and ensuring comprehensive performance assessment through diverse metrics.

Table 1. Classification of Recent Research on Intrusion Detection Systems for IIoT

Year	Authors	Technique	Models/Methods	Results
2021	M. A. Alsoufi et al.	Anomaly-Based Deep Learning	Deep Learning Models (not specified)	Systematic Literature Review
2023	S. D. A. Rihan, et al.	Ensemble Feature Selection + Deep Learning	Ensemble Learning + Deep Learning Models	Improved Detection Performance
2024	J. Li, H. Chen, et al.	Feature Reduction (FS & FE)	Decision Tree, Random Forest, Naive Bayes, k-NN, MLP	Comparison of FS & FE techniques, FE outperforms FS
2021	J. B. Awotunde, C. et al.	Deep Learning + Rule-Based FS	Deep Learning Model (not specified)	Improved Intrusion Detection
2022	B. I. Hairab, et al.	Anomaly Detection (CNN) + Regularization	CNN with Regularization Techniques	Detection of Zero-Day Attacks
2022	A. Chatterjee et al.	Anomaly Detection	Survey of Anomaly Detection Methods	Review of IoT Anomaly Detection Methods
2022	I. Ullah et al.	RNN Anomaly Detection	RNN-based Model	Anomaly Detection in IoT Networks
2022	Y. Zhang, et al.	BiLSTM DDoS Detection	BiLSTM	DDoS Attack Detection in Edge Computing
2023	H. C. Altunay et al.	Hybrid CNN+LSTM	Hybrid CNN+LSTM	Intrusion Detection in Industrial IoT Networks
2023	L. Xiaoyan and R. C. Raga	BiLSTM with Attention Mechanism	BiLSTM with Attention	Sentiment Classification (Not IIoT specific)
2023	H. Kheddar, Y. Himeur, and A. I. Awad	Deep Transfer Learning	Deep Transfer Learning Models	Intrusion Detection in Industrial Control Networks
2023	M. Mohy-Eddine, et al.	Ensemble Learning	Ensemble Learning Model	Intrusion Detection for Industrial IoT Security

III. METHODOLOGY

A. UNSW-NB15 DATASET

The UNSW-NB15 dataset was used as the benchmark for evaluating IDS in IIoT environments. This dataset was created by the Australian Centre for Cyber Security (ACCS) using the IXIA PerfectStorm tool, generating synthetic network traffic that reflects real-world scenarios, including both normal and attack behaviors. It contains 49 features derived from packet-level data and flow statistics, covering diverse attack categories such as Fuzzers, Reconnaissance, Exploits, and DoS [13].

1) Preprocessing Steps

To ensure the dataset's suitability for machine learning models, several preprocessing techniques were applied:

a) Handling Missing Values and Irrelevant Features:

Features irrelevant to classification, such as id and attack_cat, were removed. Missing values were imputed where necessary to maintain data consistency [33].

Outlier Detection: Outliers were detected using Isolation Forest, an efficient algorithm for identifying anomalies in high-dimensional data. Extreme values were clamped to the 95th percentile threshold, ensuring the data remained within a reasonable range.

Feature Transformation: Continuous features with skewed distributions were normalized using logarithmic transformation. This transformation helped reduce variability and improve model performance, particularly for deep learning algorithms that are sensitive to scale [34].

b) Feature Selection

Effective feature selection is critical to improving the efficiency and performance of IDS.

2) RFE with MI

Importance of Feature Selection in IDS: The high dimensionality of the UNSW-NB15 dataset can introduce noise and increase computational complexity. By selecting the most relevant features, the system's detection accuracy and efficiency can be significantly enhanced [23], [35].

Steps and Criteria:

1. **Feature Ranking:** Features were ranked using MI, which measures the dependency between each feature and the target variable, highlighting their predictive importance [36].

2. **Recursive Elimination:** Features were iteratively removed starting with the least significant. After each iteration, models were retrained on the remaining feature set, and performance was evaluated [37].
3. **Final Selection:** The subset of features that maximized performance while minimizing redundancy was selected. This process reduced the feature space to a manageable size without compromising accuracy.

B. DATA SPLITTING AND CROSS-VALIDATION

Robust evaluation requires an unbiased splitting of data and reliable validation techniques.

1) Stratified K-Fold Cross-Validation

The UNSW-NB15 dataset exhibits significant class imbalance, with normal traffic vastly outnumbering attack instances. Stratified K-Fold Cross-Validation was employed to ensure that each fold maintained the same class distribution as the original dataset, providing a balanced training and validation split [38].

2) Procedure:

1. The dataset was divided into K folds (10 folds), with each fold containing proportional representations of attack and normal classes.
2. The model was trained and validated iteratively on each fold, ensuring that every instance in the dataset was used for both training and validation exactly once.
3. Performance metrics were averaged across folds to provide a robust evaluation of the model's effectiveness.

C. DEEP LEARNING MODELS

IDS benefit significantly from advanced deep learning architectures capable of capturing complex patterns and temporal dependencies in network traffic. This study evaluates several deep learning models for their applicability to IDS in IIoT environments.

3) Description of Architectures

a) Convolutional Neural Network (CNN):

CNNs are employed for their strength in feature extraction, particularly in detecting spatial hierarchies within the input features. In this study, a 1D-CNN was used to analyze structured network traffic data effectively. CNNs excel in reducing dimensionality while retaining critical information [39], [40].

b) Recurrent Neural Network (RNN):

RNNs are designed to process sequential data by leveraging temporal dependencies. However, they are prone to vanishing gradient issues when handling long-term dependencies, which can limit their effectiveness for extended sequences[41].

c) Long Short-Term Memory (LSTM):

LSTM overcomes the limitations of RNN with its memory cell architecture, enabling the capture of long-range dependencies in sequential data. It has become a standard for tasks involving sequential patterns, including intrusion detection [42].

d) BiLSTM:

BiLSTM extends LSTM by processing input sequences in both forward and backward directions, providing richer contextual information. This bidirectional capability is particularly useful for identifying complex attack patterns in network traffic [30], [43].

e) Residual Neural Network (ResNet):

ResNet employs skip connections to mitigate the vanishing gradient problem, enabling the training of very deep networks. Its robustness makes it an excellent choice for feature-rich datasets like UNSW-NB15 [44].

1) Key Hyperparameters and Model Configurations

Each model was optimized with the following configurations to ensure robust performance:

- **Learning Rate:** Optimized between 10−4 and 10−3 using the Adam optimizer.
- **Batch Size:** Values of 32, 64, and 128 were tested to identify the optimal trade-off between convergence and computational efficiency.
- **Dropout Rate:** Applied at rates between 0.2 and 0.5 to mitigate overfitting.
- **Activation Functions:** ReLU for intermediate layers and softmax/sigmoid for output layers.
- **Epochs:** Models were trained for up to 50 epochs, with early stopping based on validation loss.

2) Justification for Choosing Specific Architectures

- **CNN:** Ideal for extracting spatial features, particularly effective for large-scale datasets.
- **RNN, and LSTM:** Tailored for sequential data, aligning well with the temporal nature of network traffic.
- **BiLSTM:** Its bidirectional analysis enhances the detection of complex attack patterns.
- **ResNet:** Demonstrates resilience in training deeper networks, improving feature representation.

D. COMPREHENSIVE WORKFLOW FOR ENHANCING IIOT INTRUSION DETECTION USING DEEP LEARNING MODELS

The flowchart represents a structured approach to developing a robust Intrusion Detection System (IDS) tailored for securing IIoT environments. The methodology begins with the careful preparation of the UNSW-NB15 dataset, including essential preprocessing steps like outlier handling and log transformations to ensure data quality. Feature selection is conducted using RFE with MI to identify the most relevant attributes, enhancing model efficiency. The data is then split using Stratified K-Fold Cross-Validation to ensure balanced and fair evaluation. Multiple deep learning models, such as CNN, ResNet, ANN, and BiLSTM, are trained and evaluated using comprehensive metrics, including Recall, Precision, F1-Score, and AUC. Finally, results are analyzed to identify the optimal model, with BiLSTM emerging as the standout performer, providing valuable insights for securing IIoT systems.

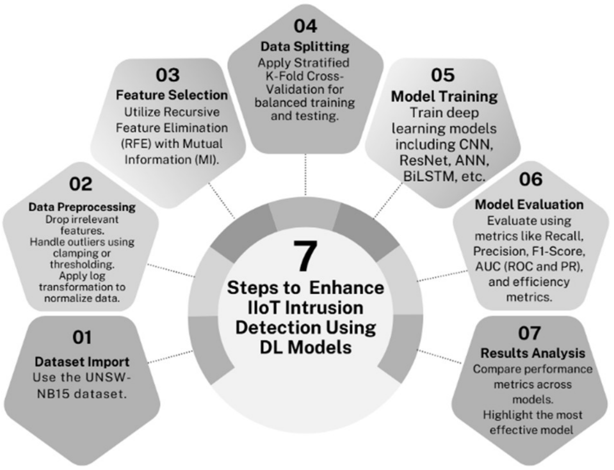

04
Data Splitting
Apply Stratified K-Fold Cross-Validation for balanced training and testing.
03
Feature Selection
Utilize Recursive Feature Elimination (RFE) with Mutual Information (MI).
05
Model Training
Train deep learning models including CNN, ResNet, ANN, BiLSTM, etc.
02
Data Preprocessing
Drop irrelevant features.
Handle outliers using clamping or thresholding.
Apply log transformation to normalize data.
06
Model Evaluation
Evaluate using metrics like Recall, Precision, F1-Score, AUC (ROC and PR), and efficiency metrics.
7
Steps to Enhance IIoT Intrusion Detection Using DL Models
01
Dataset Import
Use the UNSW-NB15 dataset.
07
Results Analysis
Compare performance metrics across models.
Highlight the most effective model

Figure **Помилка! У документі відсутній текст указаного стилю.**1. Comprehensive Workflow for Enhancing IIoT Intrusion Detection Using Deep Learning Models

IV. EXPERIMENTAL RESULTS

A.. EVALUATION METRICS

To assess the performance of the models, we used a range of metrics, each offering insights into a specific aspect of model behavior [10], [10], [12]:

- **Accuracy:** This measures how often the model correctly classified data overall.
- **Recall (Sensitivity): This** metric shows how well the model identified actual attacks. Higher recall means fewer missed detections.
- **Precision:** Precision focuses on how many of the detections were correct, helping to minimize false alarms.
- **F1-Score:** This is a balanced metric that combines Precision and Recall, providing a single value to measure overall performance.
- **ROC-AUC:** The area under the Receiver Operating Characteristic curve shows the model's ability to differentiate between attack and normal traffic.
- **MCC (Matthews Correlation Coefficient):** A balanced measure that considers all outcomes, useful for datasets with class imbalance.
- **Training Time (s):** The time taken to train the model, important for evaluating how quickly a model can be prepared for use.
- **Prediction Time (ms):** The time it takes for a model to make a prediction, crucial for real-time applications.

B. RESULTS ANALYSIS

1) Performance Across Models

From the evaluation, it was evident that the BiLSTM achieved the best results in most metrics. Its F1-Score of 0.9706 and Recall of 0.9696 demonstrate its ability to identify attacks accurately while maintaining a low false negative rate. BiLSTM also achieved the highest MCC, reflecting its robust performance across all evaluation criteria.

Other models, such as CNN and ResNet, performed well with F1-Scores of 0.9674 and 0.9612, respectively. These models were particularly effective in feature extraction, making them suitable for handling the complex patterns in network traffic.

LSTM and RNN also delivered competitive results, with F1-Scores of 0.9605 and 0.9595, respectively. However, their simpler architectures resulted in slightly lower scores compared to BiLSTM.

2) Computational Costs

While BiLSTM provided the best detection performance, it required a significantly longer training time (3505 seconds) and had the highest prediction latency (13 seconds). In comparison, RNN had the lowest computational costs, with a training time of 105 seconds and prediction latency under 1 second, making it more practical for real-time applications where computational resources are limited. Models like CNN and ResNet offered a good balance, achieving strong performance metrics while keeping training and prediction times reasonable.

C. VISUAL REPRESENTATIONS

To illustrate the findings, we used the following visualizations:

1) Performance Metrics Table

Table 2 presents a summary of key performance metrics for each model, comparing CNN, ResNet, LSTM, RNN, and BiLSTM based on various performance metrics:

- **Accuracy:** This measures the overall correctness of the model, calculated as the proportion of correct predictions out of total predictions. BiLSTM has the highest accuracy (0.967754), meaning it makes the fewest errors in classification compared to the other models.
- **Recall:** This metric shows how well the model correctly identifies positive cases (true positives). CNN and BiLSTM have the highest recall, indicating they are very good at identifying positive instances (0.969389 and 0.96961, respectively).
- **Precision:** Precision measures the accuracy of positive predictions, i.e., how many of the predicted positive cases are actually correct. ResNet has the highest precision (0.972514), meaning it is the best at minimizing false positives.
- **F1-Score:** The F1-score is the harmonic mean of precision and recall, balancing the two. BiLSTM has the highest F1-score (0.970629), reflecting its well-rounded performance in both precision and recall.
- **ROC-AUC:** This is a metric used to evaluate the ability of a model to distinguish between positive and negative classes. BiLSTM leads with the highest ROC-AUC (0.96755), showing its strong ability to separate classes correctly.
- **MCC:** The Matthews Correlation Coefficient (MCC) is another measure of classification quality that accounts for true and false positives and negatives. BiLSTM performs the best with an MCC score of 0.934886, indicating the most balanced and accurate classification across all classes.
- **Training Time (s):** This shows how long it takes for the model to train on the dataset. BiLSTM takes the longest training time (2505.54s), much more than models like ResNet (191.03s) and LSTM (164.86s), reflecting the higher complexity of BiLSTM.
- **Prediction Time (s):** This represents the time it takes for the model to make predictions after training. Again, BiLSTM has the longest prediction time (13.01s), while models like RNN and LSTM are faster (0.73s and 0.81s,

respectively), which impacts real-time usage and deployment.

Table 2. Performance Metrics of Evaluated Models

Model	Accuracy	Recall	Precision	F1-Score	ROC-AUC	MCC	Training Time (s)	Prediction Time (s)
CNN	0.9640	0.9694	0.9653	0.9674	0.9635	0.9274	414.08	1.52
ResNet	0.9579	0.9502	0.9725	0.9612	0.9587	0.9154	191.03	0.89
LSTM	0.9570	0.9521	0.9691	0.9605	0.9575	0.9135	164.86	0.81
RNN	0.9557	0.9557	0.9634	0.9595	0.9557	0.9106	105.90	0.73
BiLSTM	0.9678	0.9696	0.9717	0.9706	0.9676	0.9349	3505.54	13.01

In summary, BiLSTM outperforms the other models in accuracy, recall, precision, F1-score, ROC-AUC, and MCC, but comes with trade-offs in terms of longer training and prediction times. The choice between models depends on the specific needs, where computational efficiency might be prioritized over accuracy or vice versa.

2) Performance metrics Bar Chart

As shown in Figure 2, the bar chart compares Accuracy, Recall, Precision, and F1-Score across the models, with BiLSTM emerging as the top performer achieving the highest accuracy (96.78%), recall (96.96%), and F1-Score (97.06%), indicating the best overall balance of precision and recall.

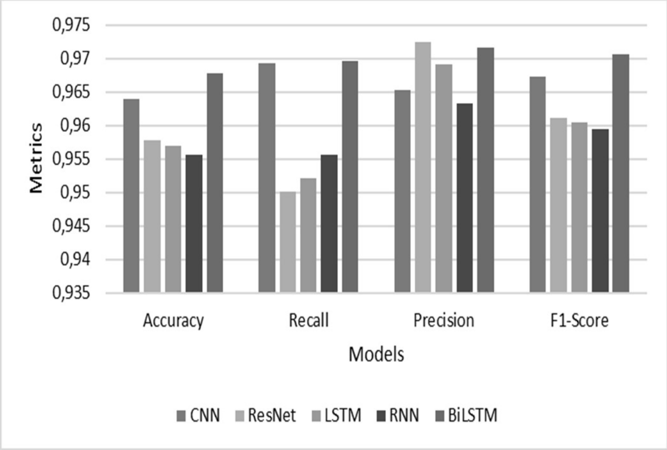


Figure **Помилка! У документі відсутній текст указаного стилю.** Comparison of Accuracy, Recall, Precision, and F1-Score Across Models

The histogram clearly highlights BiLSTM as the top-performing model across most metrics, reflecting its superior ability to balance precision and recall, which is critical in intrusion detection systems. CNN also performed well, indicating its robustness in IIoT security applications. The other models ResNet, LSTM, and RNN demonstrated competitive performance but were slightly less effective in some metrics.

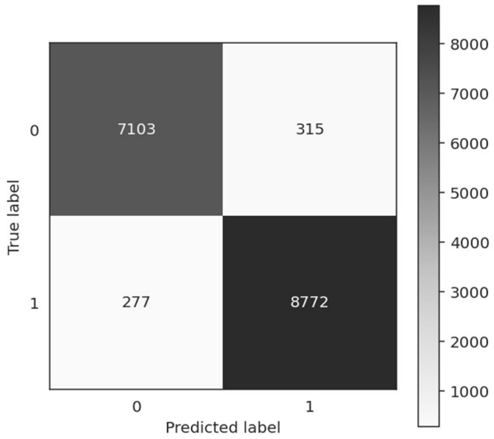
These results underscore the trade-offs between models, where BiLSTM excels in predictive power but may require higher computational resources, as discussed in the computational cost analysis.

3) Training and prediction times

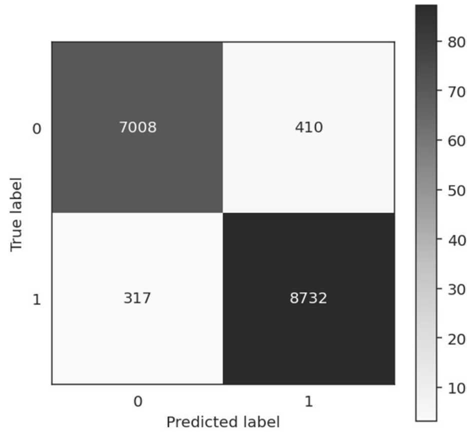
The computational costs of the models, measured in training and prediction times, show clear differences in their resource needs. The CNN took 414.08 seconds to train and 1.52 seconds to make predictions, reflecting a moderate demand. The ResNet model was faster, requiring 191.03 seconds for training and 0.89 seconds for predictions. LSTM also performed efficiently, with training taking 164.86 seconds and predictions requiring 0.81 seconds. The RNN model was the quickest, with just 0.91 seconds for training and 0.73 seconds for predictions. However, the BiLSTM, while achieving the best predictive performance, required the most time 2505.54 seconds for training and 13.01 seconds for predictions. These differences highlight the trade-offs between computational time and model complexity, which are important when choosing models for real-time IIoT intrusion detection.

4) Confusion Matrix Heatmaps:

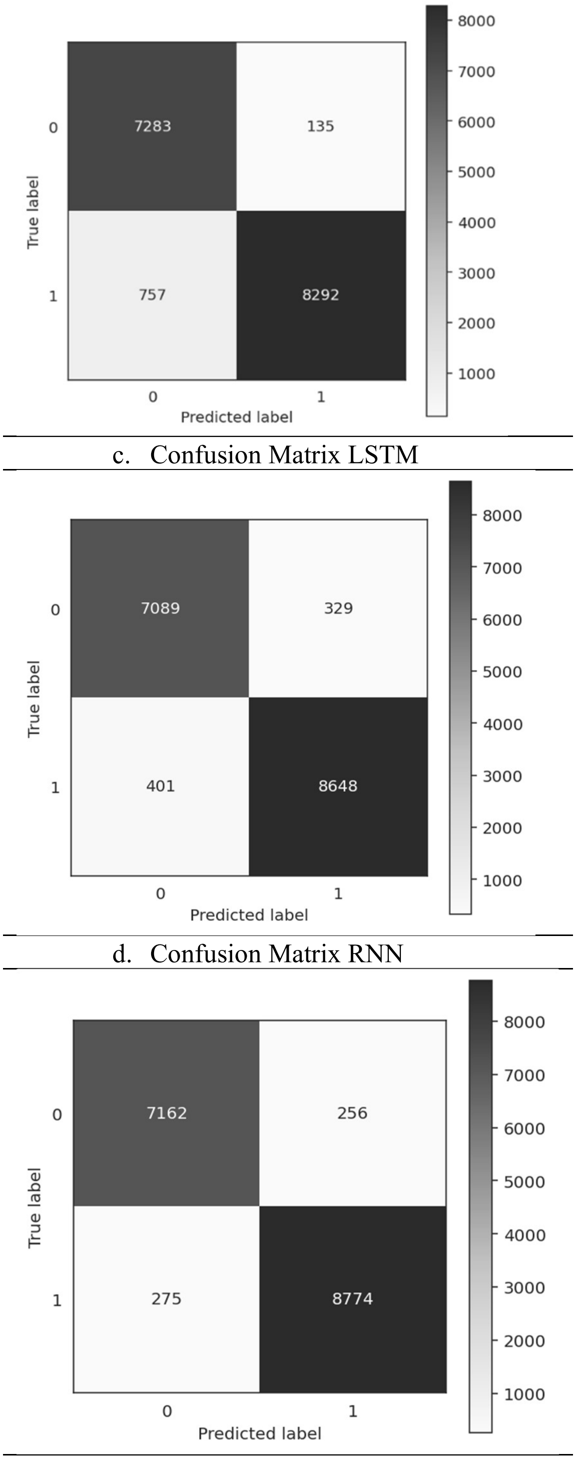
Confusion matrices for each model are visualized using heatmaps to showcase true positives, true negatives, and misclassifications, Figure 3:



a. Confusion Matrix CNN



b. Confusion Matrix ResNet



c. Confusion Matrix LSTM

d. Confusion Matrix RNN

e. Confusion Matrix BiLSTM

Figure 3. Confusion matrices for each model are visualized using heatmaps to showcase true positives, true negatives, and misclassifications: a) CNN, (b) ResNet, (c) LSTM, (d) RNN, and (e) BiLSTM.

5) *ROC-AUC and MCC Across Models*

Figure 4-2 presents a comparison of ROC-AUC and MCC across the models, showing that all models exhibit strong class-separation capabilities, with BiLSTM achieving the highest ROC-AUC score (0.96755), followed closely by CNN, ResNet, LSTM, and RNN.

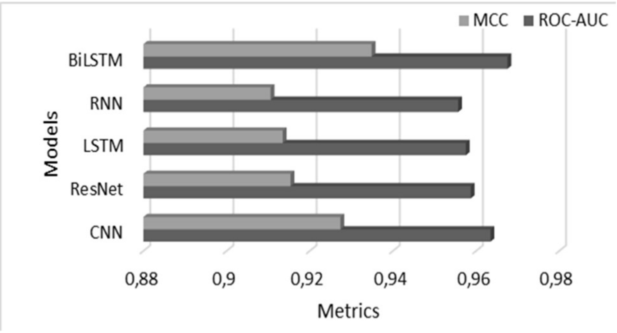


Figure **Помилка! У документі відсутній текст указаного стилю.** Comparison of ROC-AUC and MCC Across Models

For MCC, the BiLSTM model also leads with a score of 0.934886, suggesting it excels in balancing both precision and recall. The CNN (0.927363) and ResNet (0.915386) models perform very well too, but BiLSTM consistently shows the highest classification accuracy. Overall, BiLSTM outperforms the other models in both ROC-AUC and MCC.

The performance comparison across different models shows that BiLSTM consistently outperforms the other models in key metrics such as accuracy (0.967754), ROC-AUC (0.96755), and MCC (0.934886), making it the most effective model. While BiLSTM achieves superior classification performance, it comes at a cost, with significantly higher training time (2505.54s) and prediction time (13.01s) compared to models like CNN, ResNet, LSTM, and RNN, which have faster processing times. This highlights the trade-off between achieving higher accuracy and the computational efficiency of the models.

V. DISCUSSION

A. EFFECTIVENESS OF FEATURE SELECTION WITH MI

The use of RFE combined with MI proved to be a crucial step in enhancing the performance of the evaluated models. By selecting the most relevant features, the models focused on attributes that directly influenced the detection of attacks, thereby improving overall accuracy and reducing noise in the dataset. For example, the BiLSTM model achieved an F1-Score of 0.9706, demonstrating how optimized features contribute to better detection rates.

Additionally, feature selection significantly reduced the computational complexity of training. Models trained on the reduced feature set exhibited faster convergence, which is particularly beneficial for real-time applications. This approach also ensures scalability, allowing the system to adapt efficiently as the IIoT environment grows.

B. OBSERVATIONS ON DEEP LEARNING MODEL PERFORMANCE

The evaluation highlighted that advanced deep learning models outperform simpler architectures in identifying complex attack patterns. BiLSTM emerged as the most effective model, excelling in both precision (0.9717) and recall (0.9696). Its ability to process sequences in both forward and backward directions allowed it to capture temporal dependencies more effectively than other models.

Models like CNN and ResNet demonstrated strong performance in feature extraction, making them suitable for detecting spatial relationships in network traffic data. While RNN and LSTM also performed well, their simpler architectures limited their capability to match BiLSTM's effectiveness, particularly in scenarios with overlapping or complex attack patterns.

However, it is worth noting that the superior performance of BiLSTM came at a higher computational cost, with training times exceeding 3500 seconds. In contrast, RNN and CNN offered a good balance between performance and efficiency, making them viable options for scenarios where computational resources are limited.

C. BALANCING SECURITY AND COMPUTATIONAL OVERHEAD

One of the primary challenges in deploying intrusion detection systems in IIoT environments is balancing security requirements with computational constraints. While BiLSTM provided the highest detection accuracy, its long training and prediction times suggest that it may not be practical for resource-constrained or time-sensitive applications without further optimization.

On the other hand, models like CNN and RNN achieved reasonable detection rates with significantly lower computational costs. These models could be deployed on edge devices or in distributed systems, ensuring real-time processing without compromising system performance. This trade-off between accuracy and efficiency underscores the importance of tailoring model selection to the specific needs of the deployment environment.

D. APPLICABILITY TO REAL-WORLD IIOT SCENARIOS

The proposed approach demonstrates strong potential for real-world applications in IIoT environments. The use of the UNSW-NB15 dataset ensured that the models were trained and tested on realistic network traffic, including a wide variety of attack types. This makes the findings highly relevant to scenarios such as smart factories, energy grids, and industrial automation systems.

For real-time intrusion detection, lightweight models like CNN and RNN could be deployed at the network edge, while BiLSTM could serve as a secondary layer for in-depth analysis in centralized systems. This layered approach would combine the strengths of different models, balancing detection accuracy with real-time processing requirements.

Future research should explore the integration of these models with adaptive learning techniques, enabling them to handle evolving attack patterns dynamically. Additionally, optimizing BiLSTM's architecture to reduce computational costs would enhance its practicality for real-world applications.

This discussion highlights the effectiveness of feature selection, the strengths and limitations of different models, and the practical considerations for deploying intrusion detection systems in IIoT environments. It provides a solid foundation for future work aimed at advancing security in industrial systems.

VI. CONCLUSION AND FUTURE WORK

This study highlights the effectiveness of feature-driven approaches in enhancing IDS for IIoT environments. By using Recursive Feature Elimination (RFE) combined with MI for feature selection, we identified key features from the UNSW-NB15 dataset, improving detection performance while reducing computational overhead. Among the deep learning models evaluated, Bidirectional Long Short-Term Memory (BiLSTM) achieved the best results, with a recall of 96.96%, an F1-score of 97.06%, and a Matthews Correlation Coefficient (MCC) of 0.93, demonstrating its ability to capture complex temporal patterns and detect a wide variety of attacks.

However, despite BiLSTM's strong performance, its significant computational costs—especially the lengthy training time exceeding 3500 seconds—pose challenges for real-time deployment in IIoT environments. In comparison, models like Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN) exhibited slightly lower detection accuracy but offered faster training and prediction times, making them more practical for resource-constrained IIoT systems. This underscores the need to balance detection accuracy with computational efficiency when developing IDS solutions for IIoT applications.

Future research should focus on optimizing BiLSTM and other deep learning models for real-time deployment. Techniques like model pruning, quantization, and distributed training could help reduce computational demands without sacrificing performance. Hybrid models, combining the strengths of different architectures such as CNN-BiLSTM or ResNet-GRU, could further improve detection capabilities. Additionally, testing these methods on diverse datasets, including real-world IIoT traffic, is essential to ensure their generalizability. Lastly, incorporating adaptive learning mechanisms to respond to evolving cyber threats will be crucial for advancing IIoT security. This work lays the foundation for developing efficient and scalable IDS to safeguard IIoT networks against emerging cyber threats.

References

[1] O. O. Olaniyi, O. J. Okunleye, and S. O. Olabanji, "Advancing data-driven decision-making in smart cities through big data analytics: A comprehensive review of existing literature," *Current Journal of Applied Science and Technology (CJAST)*, vol. 42, no. 25, pp. 10–18, 2023, https://doi.org/10.9734/cjast/2023/v42i254181.

[2] L. Idouglid, S. Tkatek, and K. Elfayq, "Performance evaluation of deep learning models for sequence-based intrusion detection," *International Journal on Electrical Engineering and Informatics*, vol. 17, no. 1, pp. 63-77, 2025, https://doi.org/10.15676/ijeei.2025.17.1.5.

[3] M. Jouhari and M. Guizani, "Lightweight CNN-BiLSTM based intrusion detection systems for resource-constrained IoT devices," June 4, 2024, *arXiv*: arXiv:2406.02768. https://doi.org/10.1109/IWCMC61514.2024.10592352.

[4] M. Macas, C. Wu, and W. Fuertes, "A survey on deep learning for cybersecurity: Progress, challenges, and opportunities," *Computer Networks*, vol. 212, p. 109032, 2022, https://doi.org/10.1016/j.comnet.2022.109032.

[5] M. Mehmood *et al.*, "A hybrid approach for network intrusion detection," *Computers, Materials & Continua*, vol. 70, no. 1, pp. 91–107, 2022, https://doi.org/10.32604/cmc.2022.019127.

[6] M. Jouhari, H. Benaddi, and K. Ibrahimi, "Efficient intrusion detection: Combining $χ^2$ feature selection with CNN-BiLSTM on the UNSW-NB15 dataset," July 20, 2024, *arXiv*: arXiv:2407.14945. https://doi.org/10.1109/WINCOM62286.2024.10658099.

[7] P. V. Dinh, D. N. Nguyen, D. T. Hoang, Q. U. Nguyen, E. Dutkiewicz, and S. P. Bao, "Multiple-input auto-encoder guided feature selection for IoT intrusion detection systems," March 22, 2024, *arXiv*: arXiv:2403.15511. https://doi.org/10.1109/ICC51166.2024.10622942.

[8] M. Sarhan, S. Layeghy, N. Moustafa, M. Gallagher, and M. Portmann, "Feature extraction for machine learning-based intrusion detection in IoT networks," *Digital Communications and Networks*, vol. 10, no. 1, pp. 205–216, 2024, https://doi.org/10.1016/j.dcan.2022.08.012.

[9] M. Sarhan, S. Layeghy, and M. Portmann, "Feature analysis for machine learning-based IoT intrusion detection," November 23, 2022, *arXiv*: arXiv:2108.12732. https://doi.org/10.21203/rs.3.rs-2035633/v1.

[10] D. Chicco and G. Jurman, "The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation," *BMC Genomics*, vol. 21, no. 1, p. 6, 2020, https://doi.org/10.1186/s12864-019-6413-7.

[11] L. Idouglid, S. Tkatek, K. Elfayq, and A. Guezzaz, "Next-gen security in IIoT: Integrating intrusion detection systems with machine learning for

industry 4.0 resilience," *International Journal of Electrical and Computer Engineering (IJECE)*, vol. 14, no. 3, p. 3512, 2024, https://doi.org/10.11591/ijece.v14i3.pp3512-3521.

[12] Ž. Đ. Vujovic, "Classification model evaluation metrics," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 6, 2021, https://doi.org/10.14569/IJACSA.2021.0120670.

[13] N. Moustafa and J. Slay, "UNSW-NB15: a comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)," *Proceedings of the 2015 IEEE Military Communications and Information Systems Conference (MilCIS)*, Canberra, Australia: IEEE, Nov. 2015, pp. 1–6. https://doi.org/10.1109/MilCIS.2015.7348942.

[14] A. Seraj *et al.*, "Cross-validation," in *Handbook of Hydroinformatics*, Elsevier, 2023, pp. 89–105. https://doi.org/10.1016/B978-0-12-821285-1.00021-X.

[15] M. Yang and J. Zhang, "Data anomaly detection in the Internet of Things: A review of current trends and research challenges," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 14, no. 9, 2023, https://doi.org/10.14569/IJACSA.2023.0140901.

[16] V. Demertzi, S. Demertzis, and K. Demertzis, "An overview of privacy dimensions on the Industrial Internet of Things (IIoT)," *Algorithms*, vol. 16, no. 8, p. 378, 2023, https://doi.org/10.3390/a16080378.

[17] A. Heidari and M. A. Jabraeil Jamali, "Internet of Things intrusion detection systems: a comprehensive review and future directions," *Cluster Comput*, vol. 26, no. 6, pp. 3753–3780, 2023, https://doi.org/10.1007/s10586-022-03776-z.

[18] L. Idouglid, S. Tkatek, K. Elfayq, and A. Guezzaz, "A novel anomaly detection model for the Industrial Internet of Things using machine learning techniques," *Radioelectronic and Computer Systems*, no. 1, 2024, https://doi.org/10.32620/reks.2024.1.12.

[19] O. F. Awad, L. R. Hazim, A. A. Jasim, and O. Ata, "Enhancing IIoT security with machine learning and deep learning for intrusion detection," *Malaysian Journal of Computer Science (MJCS)*, vol. 37, no. 2, pp. 139–153, 2024, https://doi.org/10.22452/mjcs.vol37no2.3.

[20] M. A. Alsoufi *et al.*, "Anomaly-based intrusion detection systems in IoT using deep learning: A systematic literature review," *Applied Sciences*, vol. 11, no. 18, p. 8383, 2021, https://doi.org/10.3390/app11188383.

[21] S. D. A. Rihan, M. Anbar, and B. A. Alabsi, "Approach for detecting attacks on IoT networks based on ensemble feature selection and deep learning models," *Sensors*, vol. 23, no. 17, p. 7342, 2023, https://doi.org/10.3390/s23177342.

[22] N. B. Yusup, *Hybrid Feature Selection Technique for Classification of Human Activity Recognition*, PhD Thesis, Universiti Teknologi Malaysia, 2021.

[23] J. Li, H. Chen, M. O. Shahizan, and L. M. Yusuf, "Enhancing IoT security: A comparative study of feature reduction techniques for intrusion detection system," *Intelligent Systems with Applications*, vol. 23, p. 200407, 2024, https://doi.org/10.1016/j.iswa.2024.200407.

[24] J. B. Awotunde, C. Chakraborty, and A. E. Adeniyi, "Intrusion detection in Industrial Internet of Things network-based on deep learning model with rule-based feature selection," *Wireless Communications and Mobile Computing*, vol. 2021, no. 1, p. 7154587, 2021, https://doi.org/10.1155/2021/7154587.

[25] B. I. Hairab, M. Said Elsayed, A. D. Jurcut, and M. A. Azer, "Anomaly detection based on CNN and regularization techniques against zero-day attacks in IoT networks," *IEEE Access*, vol. 10, pp. 98427–98440, 2022, https://doi.org/10.1109/ACCESS.2022.3206367.

[26] A. Chatterjee and B. S. Ahmed, "IoT anomaly detection methods and applications: A survey," *Internet of Things*, vol. 19, p. 100568, 2022, https://doi.org/10.1016/j.iot.2022.100568.

[27] I. Ullah and Q. H. Mahmoud, "Design and development of RNN anomaly detection model for IoT networks," *IEEE Access*, vol. 10, pp. 62722–62750, 2022, https://doi.org/10.1109/ACCESS.2022.3176317.

[28] Y. Zhang, Y. Liu, X. Guo, Z. Liu, X. Zhang, and K. Liang, "A BiLSTM-based DDoS attack detection method for edge computing," *Energies*, vol. 15, no. 21, p. 7882, 2022, https://doi.org/10.3390/en15217882.

[29] H. C. Altunay and Z. Albayrak, "A hybrid CNN+LSTM-based intrusion detection system for industrial IoT networks," *Engineering Science and Technology, an International Journal*, vol. 38, p. 101322, 2023, https://doi.org/10.1016/j.jestch.2022.101322.

[30] L. Xiaoyan and R. C. Raga, "BiLSTM model with attention mechanism for sentiment classification on Chinese mixed text comments," *IEEE Access*, vol. 11, pp. 26199–26210, 2023, https://doi.org/10.1109/ACCESS.2023.3255990.

[31] H. Kheddar, Y. Himeur, and A. I. Awad, "Deep transfer learning for intrusion detection in industrial control networks: A comprehensive review," *Journal of Network and Computer Applications*, vol. 220, p. 103760, 2023, https://doi.org/10.1016/j.jnca.2023.103760.

[32] M. Mohy-Eddine, A. Guezzaz, S. Benkirane, M. Azrour, and Y. Farhaoui, "An ensemble learning based intrusion detection model for industrial IoT security," *Big Data Min. Anal.*, vol. 6, no. 3, pp. 273–287, 2023, https://doi.org/10.26599/BDMA.2022.9020032.

[33] A. E. Karrar, "The effect of using data pre-processing by imputations in handling missing values," *Indonesian Journal of Electrical Engineering and Informatics (IJEEI)*, vol. 10, no. 2, pp. 375–384, 2022, https://doi.org/10.52549/ijeei.v10i2.3730.

[34] A. Verma and V. Ranga, "Machine learning based intrusion detection systems for IoT applications," *Wireless Pers Commun*, vol. 111, no. 4, pp. 2287–2310, 2020, https://doi.org/10.1007/s11277-019-06986-8.

[35] W. Lian, G. Nie, B. Jia, D. Shi, Q. Fan, and Y. Liang, "An intrusion detection method based on decision tree-recursive feature elimination in ensemble learning," *Mathematical Problems in Engineering*, vol. 2020, pp. 1–15, 2020, https://doi.org/10.1155/2020/2835023.

[36] M. Sahaya Sheela, "Enhancing wireless sensor network security through mutual information analysis for intrusion detection and resilience," *Journal of Electrical Systems (JES)*, vol. 20, no. 5s, pp. 1957–1965, 2024, https://doi.org/10.52783/jes.2532.

[37] N. S. Yadav, V. P. Sharma, D. S. D. Reddy, and S. Mishra, "An effective network intrusion detection system using recursive feature elimination technique," *Engineering Proceedings*, vol. 59, p. 99, 2023. https://doi.org/10.3390/engproc2023059099.

[38] M. B. Musthafa *et al.*, "Optimizing IoT intrusion detection using balanced class distribution, feature selection, and ensemble machine learning techniques," *Sensors*, vol. 24, no. 13, p. 4293, 2024, https://doi.org/10.3390/s24134293.

[39] A. Halbouni, T. S. Gunawan, M. H. Habaebi, M. Halbouni, M. Kartiwi, and R. Ahmad, "CNN-LSTM: Hybrid deep neural network for network intrusion detection system," *IEEE Access*, vol. 10, pp. 99837–99849, 2022, https://doi.org/10.1109/ACCESS.2022.3206425.

[40] R. Dhahbi and F. Jemili, "A deep learning approach for intrusion detection," *International Journal of Computer Science and Network Security*, vol. 23, no. 10, pp. 89–96, 2023, doi: 10.22937/IJCSNS.2023.23.10.12.

[41] I. Ullah and Q. H. Mahmoud, "Design and development of RNN anomaly detection model for IoT networks," *IEEE Access*, vol. 10, pp. 62722–62750, 2022, https://doi.org/10.1109/ACCESS.2022.3176317.

[42] S. Shende, S. Thorat, "Long Short-Term Memory (LSTM) deep learning method for intrusion detection in network security," *International Journal of Engineering Research & Technology (IJERT)*, vol. 9, no. 6, p. IJERTV9IS061016, 2020, https://doi.org/10.17577/IJERTV9IS061016.

[43] D. V. Jeyanthi and B. Indrani, "IoT based intrusion detection system for healthcare using RNNBiLSTM deep learning strategy with custom features," *ITM Web of Conferences*, vol. 57, p. 01009, 2023. https://doi.org/10.21203/rs.3.rs-2302072/v1.

[44] A. A. Alahmadi *et al.*, "DDoS attack detection in IoT-based networks using machine learning models: A survey and research directions," *Electronics*, vol. 12, no. 14, p. 3103, 2023, https://doi.org/10.3390/electronics12143103.

Lahcen Idouglid **– Phd Student, Computer Sciences Research Laboratory, Ibn Tofail University Kenitra, Mo-rocco,**
e-mail: lahcen.idouglid@uit.ac.ma,
ORCID: 0009-0008-6570-9869,
Scopus Author ID: 57916861600.

Dr. Said TKATEK is a research professor of Computer Science at the University of Kenitra, Faculty of Science, Ibn Tofail, and a member of the Research Laboratory for Computer Science (LaRI). His current research focus is on artificial intelligence (AI), big data and their applications. He can be contacted at email: said.tkatek@uit.ac.ma.

***Khalid El Fayq** – Phd Student, Computer Sciences Research Laboratory, Ibn Tofail University Kenitra, Mo-rocco, His research focuses on the artificial intelligence, audiovisual, Software Engineering, and computer networks.*
e-mail: khalid.elfayq@uit.ac.ma,
Scopus Author ID: 57916861500.