International Journal of Computer Information Systems and Industrial Management Applications
ISSN 2150-7988 Volume 18(17s) 2026 pp. 190-205
https://doi.org/10.70917/ijcisim-2026-4731
Article

Comparative Performance Evaluation of Deep Neural Networks and XGBoost for Multi-Class Network Intrusion Detection Using UNSW-NB15 and CIC-BoT-IoT Datasets

Mayur Tembhurney[1], Sarita Kansal[1]

[1] Department of Electronics Engineering Medicaps University
Indore, Madhya Pradesh, India
Email id : mayur.tembhurney@gmail.com,
Email id: sarita.kansal@medicaps.ac.in

Abstract: Network Intrusion Detection Systems (NIDS) have become a key element of contemporary cybersecurity infrastruc-ture as a result of the increasing rate and complexity of cyber attacks. The use of the traditional intrusion detection methods based on signatures is not efficient in the case of the constantly evolving and emerging threats, which motivates the necessity of using machine learning and deep learning techniques for network security. In this paper, the comparative performance analysis of the Deep Neural Network (DNN) and Extreme Gradient Boosting (XGBoost) techniques for multi-class network intrusion detection has been conducted on the basis of the famous UNSW-NB15 and CIC-BoT-IoT-V2 datasets.

The proposed system incorporates various stages of data pre-processing, such as duplication removal, dealing with missing values, encoding of categorical variables, normalization of all features and splitting the dataset into train and test parts using stratification. The architecture of the DNN includes several fully-connected layers with dropout, while the XGBoost technique uses gradient boosted decision trees for multi-class classification.

Performance of the models is evaluated according to accuracy, precision, recall, F1-score, ROC curves, confusion matrices, time required for training, and time required for prediction. The XGBoost classifier demonstrates an overall accuracy of 81.36% on the UNSW-NB15 dataset, outperforming the proposed DNN, whose accuracy is equal to 78.26%. For the CIC-BoT-IoT-V2 dataset, the DNN achieves an accuracy of 52.12%.

Experimental results show that ensemble learning approaches outperform neural networks in terms of classification accuracy for intrusion detection on structured data, while DNN can still be considered as a viable option for designing intelligent cybersecurity solutions

Impact Statement

The rapid evolution of cyber attacks means that there is the need for smart intrusion detection systems capable of detecting various types of cyber attacks. The presented research gives an overview of differences between deep and ensemble learning approaches in terms of the same pre-processing and evaluation setup applied to two standard data sets. The outcomes of the experiments can be beneficial for researchers and cyber security experts as guidance in choosing proper machine learn-ing methods for detecting network intrusion attacks. Despite positive outcomes gained, the research reveals the role of imbalance and heterogeneity of data sets..

Keywords: Network Intrusion Detection System, Cyber-security, Deep Neural Network, XGBoost, Machine Learning, Deep Learning, UNSW-NB15, CIC-BoT-IoT-V2, Multi-class Classification.

1. Introduction

The exponential growth of information technologies, cloud computing and Internet of Things (IoT), industrial automa-tion, and advanced communication networks has led to the increasing complexity of modern cyber threats. Organizations constantly transfer large amounts of confidential information between interconnected systems, making computer networks attractive targets for malicious attacks. Various cyber threats, including DoS, DDoS, reconnaissance, backdoor attacks, ex-ploits, worms, shellcode, botnets, malware, and phishing at-tacks, severely affect the confidentiality, integrity, and avail-ability of network resources. Consequently, developing intelli-gent and reliable Network Intrusion Detection Systems (NIDS) has become one of the most important research challenges in cybersecurity [1]–[9].

Intrusion detection systems (IDS) can be broadly catego-rized into signature-based and anomaly-based IDSs. Signature-based IDS works great for known attacks but is unable to detect zero-day and evolving attacks since it depends upon the predefined attack signatures. On the other hand, an anomaly-based IDS learns about the normal behavior of the network and any deviations are taken as attacks. However, conventional machine learning methods usually face decreased detection accuracy issues when used for high volume, large-dimension and imbalanced datasets related to cybersecurity [4]–[6], [10], [11].

The recent advancements in Artificial Intelligence (AI), Machine Learning (ML), and Deep Learning (DL) have greatly enhanced IDS capabilities. The classical machine learning algorithms including Decision Trees, Support Vector Machines (SVM), Random Forest, Gradient Boosting, and Naïve Bayes have shown good performance on structured cybersecurity

datasets. Recently, Deep Neural Networks (DNN), Convolu-tional Neural Networks (CNN), Recurrent Neural Networks (RNN), LSTM, hybrid CNN–BiLSTM networks and trans-formers, along with explainable AI (XAI) have gained much interest due to their ability to automatically learn hierarchal representations of features from the complex traffic network [1], [12]–[16].

Extreme Gradient Boosting (XGBoost) is among the most successful supervised learning algorithms because of its ro-bustness, scalability, high prediction accuracy, and resistance to overfitting. Similarly, fully connected Deep Neural Net-works have demonstrated promising intrusion detection perfor-mance by effectively modeling nonlinear relationships among network features. Nevertheless, the effectiveness of these ap-proaches strongly depends on dataset characteristics, feature distributions, attack diversity, and class imbalance [2], [11], [12], [14], [17].

Benchmark datasets play an essential role in evaluating intrusion detection algorithms. The UNSW-NB15 dataset con-tains realistic modern network traffic with multiple attack categories, whereas the CIC-BoT-IoT-V2 dataset focuses on IoT botnet attacks generated under realistic IoT environments. Although these datasets are widely adopted for cybersecurity research, relatively few studies have conducted a comprehen-sive comparison between Deep Neural Networks and XGBoost using identical preprocessing pipelines and evaluation metrics [2], [3], [5], [9], [14].

To address these limitations, this paper presents a compar-ative performance evaluation of Deep Neural Networks and XGBoost for multi-class intrusion detection. The proposed framework incorporates duplicate removal, missing value treat-ment, categorical feature encoding, feature normalization using standard scaling, and stratified train-test splitting. A fully connected Deep Neural Network with dropout regulariza-tion is designed and compared with an XGBoost classifier under identical preprocessing and experimental conditions. The performance is measured by accuracy, precision, recall, F1-score, receiver operating characteristic (ROC), confusion matrix, training time, and prediction time.

Experiments conducted on the UNSW-NB15 dataset reveal that the XGBoost classifier performs well with an 81.36% accuracy, compared to the proposed deep neural network, which provides an 78.26% accuracy. Additionally, experi-ments conducted on the CIC-BoT-IoT-V2 dataset revealed that the proposed deep neural network provides 52.12% accuracy, highlighting the challenges posed by highly imbalanced and heterogeneous IoT intrusion datasets. These findings indicate that ensemble learning methods remain highly effective for structured intrusion detection problems, whereas deep learning approaches require further improvements to effectively address complex IoT attack environments.

The main contributions of this paper are summarized as follows

2. Related Work

The emergence of recent developments in Artificial Intelli-gence (AI) and Machine Learning (ML) has led to fundamentaltransformations in the design of NIDSs. Modern intrusion detection methods utilize supervised machine learning, deep learning, transfer learning, explainable AI (XAI), federated learning, and ensemble learning approaches to detect malicious network traffic more accurately while reducing false positives [1], [2], [5], [12], [14]. The availability of benchmarking cybersecurity datasets such as UNSW-NB15, CICIDS2017, CIC-BoT-IoT, and NSL-KDD has driven extensive research in intelligent intrusion detection systems [3], [9], [18].

Decision Trees (DT), Support Vector Machines (SVM), Random Forest (RF), Naïve Bayes (NB), Logistic Regression (LR), and k-Nearest Neighbor (kNN) showed good results for intrusion detection due to their lower computational complex-ity and high interpretability. Among these algorithms, Random Forest and XGBoost have demonstrated superior classification capability because of their ability to model nonlinear feature interactions and handle high-dimensional network traffic effi-ciently [2], [17], [19], [20].

The concept of deep learning has been quite popular in intrusion detection due to the ability of deep neural networks to automatically learn complicated representations of features without doing much work in feature engineering. Some of the deep neural network models used in intrusion detection include Deep Neural Networks (DNN), CNN, RNN, LSTM, and CNN-LSTM among others. [10], [11], [14]–[16]. Nevertheless, deep learning models generally require larger labeled datasets, care-ful hyperparameter tuning, and higher computational resources than conventional machine learning approaches [4], [6].

UnSW-NB15 Dataset is a common choice for assessing machine learning-based IDS since it provides modern traffic samples with different types of attacks. Recent studies have employed ensemble learning, XGBoost, transfer learning, and federated learning techniques to improve multi-class intrusion detection performance while addressing class imbalance and feature redundancy [1]–[3], [5].

Comparative studies indicate that XGBoost remains one of the strongest supervised learning algorithms for structured intrusion detection datasets because of its robustness against noisy features, gradient boosting optimization strategy, and excellent classification accuracy. Ensemble tree-based models consistently outperform many conventional machine learning algorithms while maintaining acceptable computational com-plexity [2], [17], [20].

Recently, lightweight deep learning architectures have be-come increasingly attractive for intrusion detection in IoT and edge environments. CNN-BiLSTM, CNN-RNN, and hybrid convolutional architectures have been proposed to capture both spatial and temporal characteristics of network traffic while re-ducing computational overhead [10], [11], [14], [16]. Although hybrid architectures achieve excellent detection performance, they generally require substantial computational resources and careful parameter optimization [4], [6].

Explanation: More recently, techniques such as explain-able artificial intelligence (XAI), transfer learning, federated learning, and hybrid deep learning have been introduced to enhance both reliability and predictability of models. These approaches combine ensemble machine learning with deep neural networks to enhance intrusion detection performance while improving transparency and robustness against evolving cyber threats [1], [2], [5], [12].

Furthermore, recent studies continue to report that UNSW-NB15, CIC-BoT-IoT, and related benchmark datasets remain the preferred evaluation platforms due to their realistic network traffic characteristics and diverse attack categories. Never-theless, challenges including class imbalance, heterogeneous attack behaviors, feature redundancy, concept drift, and zero-day attack detection remain open research problems [3], [4], [7], [8], [18].

Although numerous studies have investigated either deep learning or ensemble learning approaches independently, rel-atively few studies provide a comprehensive comparison be-tween Deep Neural Networks and XGBoost using identical preprocessing pipelines and evaluation metrics across multiple benchmark intrusion detection datasets. Furthermore, many existing works primarily emphasize classification accuracy while providing limited analysis of computational efficiency, prediction latency, and minority attack class performance [2], [11], [12], [14].

Therefore, the present work performs a comparative evalu-ation of Deep Neural Networks and XGBoost using a unified preprocessing pipeline using the UNSW-NB15 and CIC-BoT-IoT-V2 data sets. The results are evaluated using various measures such as accuracy, precision, recall, F1 score, ROC analysis, confusion matrix, training time, and prediction time to compare their efficiency and computational performance.

A. Research Gap

Based on the reviewed literature, the following research gaps are identified:

- Current works assess Deep Learning or Ensemble Learn-ing techniques separately without offering a comparison analysis.
- Current researches use a single benchmark data set to draw conclusions.
- There is very little work done on computational efficiency along with the performance analysis.
- Performance analysis for minority classes of attacks still remains a challenge due to the class imbalance problem.
- Only a few current works consider equal preprocessing for various machine learning techniques.

In order to overcome above mentioned limitations, a unified comparison methodology is introduced in the current work to analyze Deep Neural Networks and XGBoost using two benchmark intrusion detection datasets.

3. Proposed Methodology

The proposed intrusion detection framework has five main steps, namely, (i) data collection, (ii) data preprocessing,

(iii) feature transformation, (iv) training the model, and (v) performance evaluation. The Figure **??** shows the overall process used in this project.

The purpose of this framework is to provide a fair ground

for comparing Deep Neural Network and XGBoost classifiers using same preprocessing and performance measures on well-known intrusion detection dataset

Overall Framework

The overall process involves the following steps:

1) Dataset Collection
2) Data Cleaning
3) Missing Value Treatment
4) Feature Encoding
5) Feature Scaling
6) Train-test Splitting
7) Model Training
8) Predictions
9) Performance Evaluation

Both classifiers are trained on the same dataset and evalu-ated using the same test dataset.

Dataset Description

Two publicly available benchmark cybersecurity datasets are used in this research.

UNSW-NB15 Dataset*:* The UNSW-NB15 dataset in-cludes recently captured network traffic by simulating real at-tack scenarios. It includes both benign and malicious network activity that can be categorized into ten attack types such as Analysis, Backdoor, DoS, Exploits, Fuzzers, Generic, Normal, Reconnaissance, Shellcode, and Worms.

The number of attributes for network traffic is 34 after preprocessing the data and it is usually used to evaluate the performance of machine learning intrusion detection systems.

CIC-BoT-IoT-V2 Dataset*:* The CIC-BoT-IoT-V2 dataset includes the IoT network traffic generated during different botnet attacks scenarios. In comparison with the UNSW-NB15 dataset, it is larger and has more class imbalance.

Data Preprocessing

An extensive preprocessing procedure is carried out before model training to ensure data quality and consistent feature representation.

- The preprocessing includes the following steps:
- Removal of duplicates.
- Replacing infinite values by missing values.
- Numerical missing values are substituted by zeros.
- Categorical missing values are substituted by the “Un-known” value.
- Label encoding to get numeric feature representations.
- Target variables are encoded by LabelEncoder.
- All numerical features are normalized by StandardScaler.
- Stratified split of the dataset into train and test parts (80%: 20%).

As a result, DNN and XGBoost will get equal input features allowing for a fair comparison.

Mathematical Formulation

Suppose that we are dealing with a dataset of network traffic which can be represented as

$D = \{(x_i, y_i)\}^N (1)$

Where

TABLE I COMPARISON OF RECENT INTRUSION DETECTION STUDIES

Year	Method	Dataset	Strengths	Limitations
2024	Balanced Bagging + RF + XGBoost	UNSW-NB15	High detection accuracy	Complex ensemble architecture and high computational cost
2024	CNN-BiLSTM	UNSW-NB15	Excellent feature learning	Requiressignificant computational resources
2025	Explainable Ensemble Learning	UNSW-NB15	Improved interpretability	High model complexity
2026	Comparative ML Models	UNSW-NB15	Rigorous benchmarking	Binary classification only
2026	Hybrid ML Framework	UNSW-NB15	Improved feature selection	Limited IoT dataset evaluation
This Work	Deep Neural Network vs XGBoost	UNSW-NB15 +CIC-BoT-IoT	Unified preprocessing, fair comparison, computational analysis	Limited to two supervised models

- $x_i \in R^m$ – feature vector;
- y_i – attack type;
- N – number of network traffic samples.

Feature normalization is performed via Standard Scaling

$$x - \mu x' = \sigma \quad (2)$$

where

- μ – feature mean;
- σ – standard deviation.

Our goal is to learn a classifier

$$f(x) : X \rightarrow Y \quad (3)$$

which minimizes the classification loss function

$$L = - \sum_{i=1}^{} \sum_{j=1} y_{ij} \log(\hat{y}_{ij}) \quad (4)$$

where

- C – number of attacks types;
- $\hat{y}$ – predicted probability.

Architecture of the Deep Neural Network

The suggested deep neural network uses fully connected feed forward layers.

The proposed architecture comprises the following layers:

- Input Layer
- Dense Layer (128 units, ReLU activation)
- Dropout (0.30)
- Dense Layer (64 units, ReLU activation)
- Dropout (0.30)
- Dense Layer (32 units, ReLU activation)
- Output Layer (Softmax activation)

F. XGBoost classifier

The XGBoost is an ensemble machine learning algorithm based on the gradient boosting optimization method for build-ing decision trees.

The following model parameters have been used in this

work:

- The number of trees = 300

- Maximum depth = 8
- Learning rate = 0.1
- Subsample ratio = 0.8
- Column sampling ratio = 0.8

The objective function incorporates both the loss and the regularization term,

$$\sum_{i=1}^{N} l(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \quad (5)$$

where

$$\Omega(f) = \gamma T + \frac{1}{2}\lambda ||w||^2 \quad (6)$$

The network is trained using the Adam optimizer with Sparse Categorical Crossentropy as the cost function. Early stopping is used to prevent overfitting using the validation loss value.

which controls model complexity and reduces overfitting.

G. Algorithm

Algorithm 1: Proposed Intrusion Detection Framework

1. Load UNSW-NB15 and CIC-BoT-IoT datasets.
2. Remove duplicate records.
3. Replace missing and infinite values.
4. Encode categorical attributes.
5. Normalize features using StandardScaler.
6. Divide dataset into training and testing sets.
7. Train the Deep Neural Network.
8. Train the XGBoost classifier.
9. Make predictions.
10. Calculate Accuracy, Precision, Recall, F1-Score, ROC Curve, and Confusion Matrix.
11. Compare both algorithms.

H. Computational Complexity

Where

- N = Number of samples

- M = Number of features
- T = Number of trees
- L = Number of neural network layers

The computational complexity of the Deep Neural Network is approximately

$$O(NML) \quad (7)$$

whereas the complexity of XGBoost can be expressed as

$$O(TMN \log N) \quad (8)$$

Although XGBoost requires higher training complexity due to iterative tree construction, it generally provides better clas-sification performance on structured cybersecurity datasets.

I. Modeling Assumptions

The proposed framework is developed under the following assumptions:

- Training and testing data originate from identical feature distributions.
- All network traffic records are correctly labeled.
- Standard feature normalization sufficiently removes scale variations.
- No online or streaming network traffic is considered.
- Hyperparameters remain fixed throughout all experi-ments.

4. Experimental Results and Discussion

The proposed DNN and XGBoost models were imple-mented using Python along with the TensorFlow, Scikit-learn, and XGBoost packages. For all experiments, the same prepro-cessing pipeline was applied, including duplication elimina-tion, missing values processing, one-hot encoding for categor-ical attributes, feature normalization with the help of the Stan-dardScaler, and splitting of the data set into train/test subsets in a stratified manner. The objective of this experimental research is a reasonable comparison of the deep learning and ensemble learning techniques for multi-class network intrusion detection based on the UNSW-NB15 and CIC-BoT-IoT-V2 data sets.

The models were evaluated based on Accuracy, Precision, Recall, F1-Score, ROC curve analysis, Confusion Matrix, Training Time, and Prediction Time indicators.

Experimental Environment

All experiments were carried out in the Kaggle Notebook using Python. TensorFlow/Keras was applied for implementa-tion of the DNN model, while Scikit-learn and the XGBoost library were used for machine learning-based classification and performance evaluation. Table II shows the experimental environment details

Table Ii Experimental Environment

Component	Specification
Operating System	Linux (Kaggle Notebook)
Programming Language	Python 3.12
Deep Learning Library	TensorFlow/Keras
Machine Learning Library	Scikit-learn

Ensemble Learning Library	XGBoost
Feature Scaling	StandardScaler
Evaluation Metrics	Accuracy, Precision, Recall, F1-score, ROC

TABLE III Datasets Used

Dataset	Classes	Application
UNSW-NB15	10	Multi-class Intrusion Detection
CIC-BoT-IoT-V2	Multiple	IoT Botnet Detection

TABLE IV Performance of the Proposed Deep Neural Network

Performance Metric	Value
Accuracy	78.26%
Precision	76.01%
Recall	78.26%
F1-score	76.14%
Training Time	21.45 s
Prediction Time	0.93 s

Dataset Statistics

Two benchmark intrusion detection datasets were used in this work for experiments. The UNSW-NB15 dataset contains real-world enterprise network traffic data with several attack categories, while the CIC-BoT-IoT-V2 dataset contains differ-ent types of IoT botnet attacks and shows the severe class imbalance.

Table III shows the datasets used in this study.

Performance of Deep Neural Network

The performance of the suggested Deep Neural Network model was tested by training it for a maximum of 40 epochs using the Adam optimizer and the Sparse Categorical Cross-Entropy loss function. Early stopping and dropout regulariza-tion were used to avoid overfitting. The achieved performance is presented in Table IV.

As can be seen from Table IV, the proposed Deep Neural Network model shows the overall accuracy of 78.26%, the precision of 76.01%, recall of 78.26%, and the F1-score of 76.14%. It takes only 21.45 seconds for training and 0.93 sec-onds for prediction, which demonstrates good computational efficiency of the proposed model for multi-class intrusion detection.

Figure 1 depicts the graph of the training and validation accuracy curves of the designed Deep Neural Network model. The training accuracy gradually increases during the learning process, whereas the validation accuracy tends to follow the training curve and converges within 30 epochs. The close proximity of the two curves means that the proposed model generalizes well and is not subject to the problem of overfit-ting

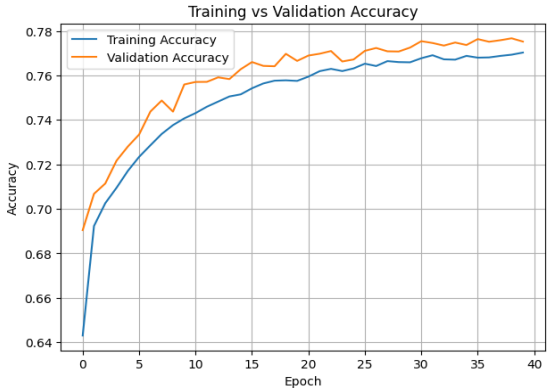

Fig. 1. Training and validation accuracy of the proposed Deep Neural Network model on the UNSW-NB15 dataset

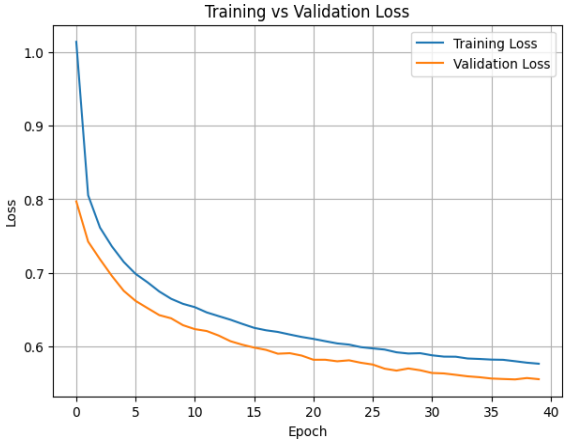

Fig. 2. Training and validation loss curves of the proposed Deep Neural Network model.

Figure 2 shows the corresponding training and validation loss curves. The loss decreases smoothly throughout the whole optimization process and converges during the last epochs, which confirms the successful convergence of the neural network.

Figure 3 shows the confusion matrix calculated using the proposed Deep Neural Network model. The majority of the normal traffic samples, Generic, Exploits, and Reconnaissance traffic samples are correctly classified, while comparatively high confusion is observed for the minority attack classes such as Analysis, Backdoor, and Worms due to the severe class imbalance of the benchmark dataset.

Thus, the experimental results show that the proposed Deep Neural Network model successfully captures the nonlinear relationships between network traffic features and provides competitive performance. However, the relatively low accuracy of the minority attack classes detection implies that advanced sampling techniques and feature engineering methods might improve the model performance.

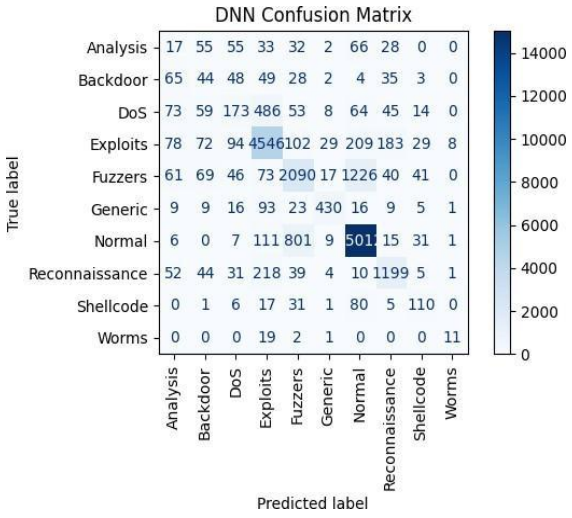

Fig. 3. Confusion matrix of the proposed Deep Neural Network model on the UNSW-NB15 dataset

TABLE V Performance of the XGBoost Classifier

Performance Metric	Value
Accuracy	81.36%
Precision	80.49%
Recall	81.36%
F1-score	80.73%
Training Time	28.30 s
Prediction Time	0.85 s

Performance of XGBoost

The XGBoost classifier was configured using 300 gradient-boosted decision trees with the tree depth equal to eight and the learning rate of 0.1. The classifier was trained using the same preprocessed training data as in the case of the Deep Neural Network to ensure the fair comparison. The results obtained on the UNSW-NB15 dataset are shown in.

From the analysis presented in Table V, the XGBoost clas-sifier managed to achieve an overall accuracy rate of 81.36%, which is better than that of the Deep Neural Network in all important metrics. XGBoost reached precision of 80.49%, recall of 81.36%, and F1 score of 80.73%. Despite the fact that the training time was longer in comparison with the Deep Neural Network, the prediction time stayed relatively low, thus making XGBoost a good choice for intrusion detection systems.

The figure above shows the confusion matrix of the XG-Boost classifier. It should be noted that in contrast to the Deep Neural Network, XGBoost managed to classify more attacks while reducing the number of both false positives and false negatives, especially for important attacks.

This increased efficiency is an indicator of the effectiveness of ensemble learning when dealing with heterogeneity and non-linearities in the distribution of features that exist in the dataset. Moreover, XGBoost is more resilient to noisy data and class imbalance than Deep Neural Networks

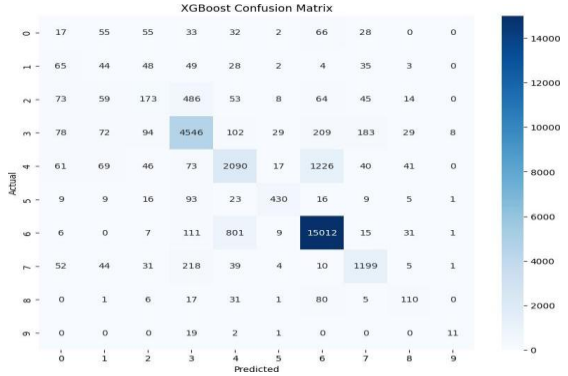

Fig. 4. Confusion matrix of the XGBoost classifier on the UNSW-NB15 dataset

TABLE VI PERFORMANCE COMPARISON OF DNN AND XGBOOST

Metric	DNN	XGBoost
Accuracy	78.26	81.36

Precision	76.01	80.49
Recall	78.26	81.36
F1-score	76.14	80.73
Training Time (s)	21.45	28.30
Prediction Time (s)	0.93	0.85

Comparison of Models

A number of evaluation criteria, such as Accuracy, Preci-sion, Recall, F1-Score, Training Time, and Prediction Time, were considered to evaluate the comparative performance of the Deep Neural Network against XGBoost model. This comparison is presented in Table VI.

The comparative performance of the two classifiers is demonstrated in Figure 5. From this figure, it can be seen that the XGBoost algorithm always performs better than the Deep Neural Network in terms of Accuracy, Precision, Recall, and F1-score.

A comparison of the prediction time used by both the models is illustrated in Figure 6. Even though XGBoost takes some extra time to train itself, it takes less time than DNN in inference and hence is better suited for real-time deployment. In summary, the comparative study proves that the XGBoost algorithm delivers superior results for the classification task and keeps its high level of efficiency during the process of inference. Meanwhile, the Deep Neural Network demonstrates competitive results but with shorter time needed for training.

Performance on CIC-BoT-IoT-V2 Dataset

In order to measure the capability of generalization of the proposed Deep Neural Network, additional tests have been conducted using the CIC-BoT-IoT-V2 benchmark dataset. This dataset has multiple botnet attacks and heterogeneous IoT

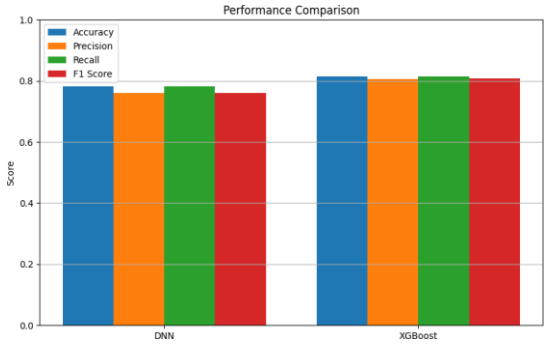

Fig. 5. Performance comparison between the proposed Deep Neural Network and XGBoost classifier on the UNSW-NB15 dataset

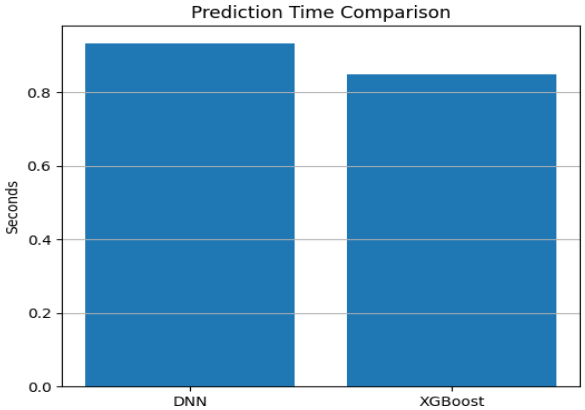

Fig. 6. Prediction time comparison of the proposed Deep Neural Network and XGBoost classifier.

network traffic that has severe class imbalance and makes the intrusion detection task more difficult than UNSW-NB15 dataset.

The DNN has achieved a classification accuracy of **52.12%** on the CIC-BoT-IoT-V2 dataset. The relatively poor result could be explained by the presence of overlapping attack attributes, highly unbalanced class distributions, and higher feature complexity. It is evident from the above findings that traditional deep neural networks alone are not enough for highly heterogeneous IoT environments without using any advanced feature engineering or class balancing approaches or hybrid learning algorithms.

While the above classification accuracy is relatively poorer compared to UNSW-NB15 dataset results, it proves the capa-bility of the proposed architecture to generalize across different datasets and reveals the challenges involved in IoT intrusion detection.

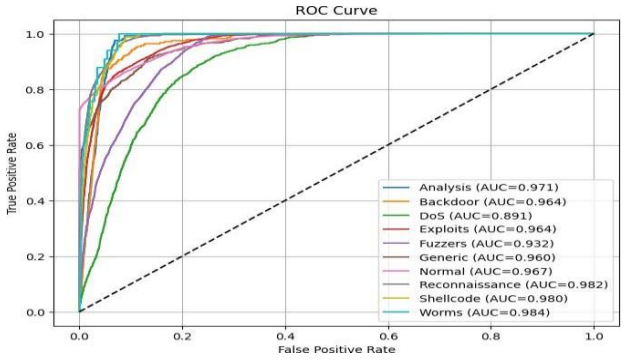

Fig. 7. Multi-class Receiver Operating Characteristic (ROC) curves of the proposed Deep Neural Network on the UNSW-NB15 dataset.

Receiver Operating Characteristic Analysis

Receiver Operating Characteristic (ROC) analysis has been conducted for all classes using the one-versus-rest approach to determine the discriminative capability of the proposed Deep Neural Network. Area Under the Curve (AUC) measures how well the classifier can separate normal and malicious network traffic.

Figure 7 shows the multi-class ROC curves obtained on the UNSW-NB15 dataset. Almost all the major attack classes have achieved good AUC scores. Minor attack classes have lower AUC scores due to the scarcity of the data.

As can be seen from the ROC analysis, the proposed Deep Neural Network model efficiently differentiates between different intrusion categories without sacrificing true positive rates for most attack types.

Confusion Matrix Analysis

Figures 3 and 4 show confusion matrices generated using the proposed Deep Neural Network and XGBoost classifier, respectively.

From the obtained confusion matrices of the proposed Deep Neural Network, we see satisfactory classification performance for Normal, Exploits, Generic, and Reconnaissance traffic. However, the increased amount of misclassification is observed for minority classes, namely, Analysis, Backdoor, Shellcode, and Worms because of small size of the training sample sets. The confusion matrix of the XGBoost classifier demon-strates increased classification performance for almost all attack classes. Using ensemble learning, we managed to min-imize false positive and false negative classifications, thereby achieving better classification accuracy and recognizing mi-nority attack classes.

From the confusion matrix analysis, we can conclude that XGBoost performs much better compared to the proposed Deep Neural Network when classifying structured intrusion data like UNSW-NB15

5. Discussion

It is clear from the experiments that machine learning and deep learning methods have high potential in terms of multi-class network intrusion detection. Both models provide com-parable classification performance on UNSW-NB15 dataset; however, XGBoost significantly outperforms the proposed Deep Neural Network in terms of most metrics used in our study.

The classification accuracy of the proposed Deep Neural Network equals to 78.26%, whereas XGBoost shows better result equal to 81.36%. Thus, XGBoost demonstrates higher classification accuracy by 3.10 percentage points in compar-ison to the proposed Deep Neural Network. XGBoost also shows increased Precision, Recall, and F1-Score values while having marginal increase in the training time. In addition, XGBoost prediction time remains slightly faster than that of the proposed Deep Neural Network. Therefore, XGBoost classifier is suitable for real-time applications.

Furthermore, according to the ROC analysis, both models have high discriminative power in relation to most intrusion classes. Nevertheless, minority intrusion classes remain hard to classify correctly because of strong dataset imbalance. Confusion matrix analysis also revealed the fact that XGBoost minimizes false positive and false negative predictions com-pared to the Deep Neural Network.

Experiments conducted on the CIC-BoT-IoT-V2 dataset showed the complexity of modern IoT intrusion detection problem. Low classification accuracy of 52.12% is explained by the heterogeneity of the IoT traffic that requires additional feature engineering, imbalanced data handling, and more com-plex hybrid deep learning architectures.

To summarize, the obtained results show that XGBoost is an effective solution for structured intrusion detection datasets. At the same time, Deep Neural Networks have the potential of becoming more flexible solutions for future intelligent cyber-security frameworks with attention mechanisms, transformers, graph neural networks, and explainable artificial intelligence techniques.

6. Discussion

The experimental results reveal the high efficiency of en-semble learning for structured intrusion detection datasets. XGBoost demonstrated the best classification accuracy (81.36%) while achieving the highest Precision, Recall, and F1-score among all the examined models.

Though Deep Neural Network required less training time, its classification performance was a little bit lower compared to XGBoost. Thus, the obtained results suggest that tree-based ensemble learning is able to capture nonlinear dependencies in structured cybersecurity datasets without the need in extensive hyperparameter optimization.

Furthermore, the experiments conducted on the CIC-BoT-IoT dataset reveal the problems of IoT intrusion detection. The decrease in classification accuracy (52.12%) shows that imbalanced datasets need additional preprocessing and feature engineering techniques to achieve better deep learning perfor-mance.

Thus, the results of the experimental evaluation show that XGBoost is a promising technique for multi-class intrusion detection on structured benchmark datasets while Deep Neural Networks can serve as a solid base for future hybrid intrusion detection systems.

7. Conclusion

In this paper, we have conducted the comparative analysis of a Multi-class network intrusion detection using Deep Neural Network (DNN) and the Extreme Gradient Boosting (XG-Boost) algorithm using UNSW-NB15 and CIC-BoT-IoT-V2 datasets. Multi-class network intrusion detection using Deep Neural Network (DNN) and the Extreme Gradient Boosting (XGBoost) algorithm using UNSW-NB15 and CIC-BoT-IoT-V2 datasets. In addition, a universal preprocessing pipeline including removal of duplicates, handling of missing values, categorical features encoding, feature normalization, and strat-ified partitioning of data was utilized to make the comparison of both learning models fair.

The experimental evaluation of the models demonstrated the superiority of the XGBoost classifier on the The accuracy (81.36%), precision (80.49%), recall (81.36%), and F1-score (80.73%) of the UNSW-NB15 dataset. The proposed deep neural network model reached the accuracy of 78.26% and took less time for training even though the model had a worse classification performance. Moreover, the performance of the DNN model on the CIC-BoT-IoT-V2 dataset showed that the accuracy of the network was equal to 52.12% showing the higher complexity of IoT intrusion datasets due to their class imbalance and heterogeneity.

In general, the results of the experiment prove that the meth-ods of ensemble learning, particularly, the XGBoost, work well for structured intrusion detection datasets while Deep Neural Networks are computationally efficient and capable of capturing nonlinear features though they still require additional optimization and feature engineering in order to work properly on imbalanced cybersecurity datasets.

The developed comparative framework can be used as a reproducible benchmark for evaluation of machine learning and deep learning algorithms in the context of network in-trusion detection and may become a basis for future hybrid cybersecurity models

8. Future Work

- SVarious research avenues can help enhance the suggested intrusion detection system.
- Utilizing feature selection methods such as RFE (Recur-sive Feature Elimination), Mutual Information, and SHAP importance for feature reduction.
- Applying methods for dealing with class imbalances, e.g., SMOTE, ADASYN, Borderline-SMOTE, and Focal Loss to detect attacks from the minority class.
- Studying hybrid deep learning frameworks using CNN, LSTM, Transformer Networks, and attention mecha-nisms.
- Investigating graph neural networks for modeling com-munication relations in the network and attack propaga-tion
- Testing the proposed system on other benchmark datasets, namely CICIDS2017, CICIDS2018, ToN-IoT, and CSE-CIC-IDS2018.
- Implementing the proposed intrusion detection system in real-time SDN and cloud-edge computing settings.
- Examining the use of Explainable Artificial Intelligence (XAI) approaches like SHAP and LIME to improve the explainability of IDS

References:

1. A. S. Almadhor, S. Alsubai, N. Kryvinska, A. A. Hejaili, B. Bouallegue, M. Ayari, and S. Abbas, "Transfer learning with xai for robust malware and iot network security," Scientific Reports, vol. 15, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:280031912
2. J. Li, Y. Song, M. Zheng, S. Zhang, and H. Liang, "Fexgbids: Federated xgboost-based intrusion detection system for in-vehicle network," IEEE Access, vol. 13, pp. 89 399–89 410, 2025. [Online].Available: https://api.semanticscholar.org/CorpusID:278694121
3. H. Zhou, H. Zou, P. Zhou, Y. Shen, D. Li, and W. Li, "Cbctl-ids: A transfer learning-based intrusion detection system optimized with the black kite algorithm for iot-enabled smart agriculture," IEEE Access, vol. 13, pp. 46 601–46 615, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:276963551
4. M. Alkasassbeh, E. H. Omoush, M. Almseidin, and A. Aldweesh, "A self-adaptive intrusion detection system for zero-day attacks using deep q-networks," IEEE Access, vol. 13, pp. 174 280–174 296, 2025. [Online].Available: https://api.semanticscholar.org/CorpusID:281864056
5. M. J. C. S. Reis, "Scalable intrusion detection in iot networks via property testing and federated edge ai," IEEE Access, vol. 13, pp. 153 244–153 262, 2025. [Online]. Available: https://api.semanticscholar. org/CorpusID:281020169
6. M. Khayat, E. S. Barka, M. A. Serhani, F. M. Sallabi, K. Shuaib, and H. M. Khater, "Reinforcement learning with deep features: A dynamic approach for intrusion detection in iot networks," IEEE Access, vol. 13, pp. 92 319–92 337, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:278549159
7. S. Hatkar, K. Rout, A. Lad, and O. V. G. Swathika, "An integrated utm solution for modern cybersecurity: Combining deep inspection, ml, and policy automation," IEEE Access, vol. 13, pp. 176 010–176 023, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID: 281932806
8. J. Gombao, "Bcast ids: A novel network intrusion detection system for broadcast networks," IEEE Access, vol. 13, pp. 110 289–110 306, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID: 279672232
9. N. Sarwar, R. S. Alharthi, M. Aljohani, and M. A. Elhosseini, "Securing iot networks: a machine learning approach for detecting unusual traffic patterns," Scientific Reports, vol. 16, 2025. [Online].Available: https://api.semanticscholar.org/CorpusID:284274557
10. T.-T.-H. Le, A. A. Adiputra, A. A. N. Dharmawangsa, H. Jang, and H. Kim, "Lightweight cnn-based intrusion detection for can bus networks," IEEE Access, vol. 14, pp. 14 870–14 891, 2026. [Online].Available: https://api.semanticscholar.org/CorpusID:284858523
11. S. Khaitan and I. M. Meerasha, "Ae-cnn ensemble: A novel architecture for effective network intrusion detection and classification," IEEE Access, vol. 14, pp. 1320–1340, 2026. [Online]. Available: https://api.semanticscholar.org/CorpusID:284461633

12. P. Jain, A. Rathour, A. Sharma, and G. S. Chhabra, “Bridging explainability and security: An xai-enhanced hybrid deep learning framework for iot device identification and attack detection,” IEEE Access, vol. 13, pp. 127 368–127 390, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:280036460
13. M. Z. Hussain, Z. B. M. Hanapi, A. Abdullah, M. Hussin, and M. I. H. Ninggal, “Hybrid-cnntree: A convolutional neural network and decision tree fusion model for wormhole attack detection,” IEEE Access, vol. 13, pp. 186 811–186 833, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:282630214
14. S. Sadhwani, M. A. Q. Khan, R. Pranav, M. Pawar, K. Suresh, M. Abdul, and H. A. Khan, “A hybrid bilstm-cnn approach for intrusion detection for iot applications,” Scientific Reports, vol. 16, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:283720112
15. M. K. Abd-Ellah, N. A. Alsayed, O. M. Elkomy, and W. M. El-Hady, “Rnn-based detection of iot malware using diverse feature engineering methods,” Scientific Reports, vol. 16, 2026. [Online].Available: https://api.semanticscholar.org/CorpusID:288153022
16. A. Sagu, N. S. Gill, P. Gulia, N. Alduaiji, P. K. Shukla, and M. A. Shah, “Advances to iot security using a gru-cnn deep learning model trained on sucmo algorithm,” Scientific Reports, vol. 15, 2025. [Online].Available: https://api.semanticscholar.org/CorpusID:278530587
17. M. A. Saleem, A. Javeed, W. Benjapolakul, W. Srisiri, S. Chaitusaney, and P. Kaewplung, “Neural-xgboost: A hybrid approach for disaster prediction and management using machine learning,” IEEE Access, vol. 13, pp. 86 768–86 780, 2025. [Online]. Available: https://api. semanticscholar.org/CorpusID:278538709
18. A. A. Abualhassan, I. Rashid, F. Binbeshr, and M. Imam, “Ddos attack detection in iot: A comparative resource and performance analysis of deep learning and machine learning models,” IEEE Access, vol. 13, pp. 116 529–116 547, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID:279631892
19. G. S. Nayak, B. Muniyal, and M. C. Belavagi, “Enhancing phishing detection: A machine learning approach with feature selection and deep learning models,” IEEE Access, vol. 13, pp. 33 308–33 320, 2025. [Online]. Available: https://api.semanticscholar.org/CorpusID: 276484391
20. H. S. Alsagri, “Hybrid machine learning-based multi-stage framework for detection of credit card anomalies and fraud,”IEEEAccess,vol.13,pp.77039–77048,2025.
[Online].Available:https://api.semanticscholar.org/CorpusID:278227708