IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

Deep Learning-Based Intrusion Detection System for High-Speed Networks

Kiran Yadav[1] and Anuradha Pathak[2]
Research Scholar, Dept of Electronics & Communication[1]
Asst. Prof., Dept of Electronics & Communication[2]
NITM, Gwalior India
kiran085@gmail.com, anuradha.pathak27@gmail.com

Abstract: *The rise in IoT devices and their diverse applications has heightened the importance of IoT security. Research on network security indicates that Distributed Denial of Service (DDoS) attacks on IoT systems are becoming more frequent, advanced, and varied. DDoS attacks have evolved into serious cyber threats, enabling lucrative and efficient cybercrimes. Among the most hazardous risks to network security, DDoS attacks present significant challenges for machine learning (ML)-based detection systems, often impacting their accuracy. Artificial intelligence (AI), which integrates ML for cyberattack detection, is the most widely used approach in this domain. This study proposes a model for identifying and reducing DDoS attacks in Software-Defined Networking (SDN) using ML techniques. The model compares the F1-score, recall, accuracy, and precision of various ML algorithms, incorporating Extra Tree and Cat Boost classifiers. To enhance detection capabilities, DDoS-Net effectively addresses data imbalance and incorporates a comprehensive feature analysis.The evaluation of DDoS-Net on the UNSW-NB15 dataset highlights its outstanding performance. The most significant level of accuracy attained using Cat Boost and Extra Tree classifiers is 90.78%, 90.27%, respectively. This research introduces a robust and accurate method for detecting DDoS attacks, significantly enhancing cyber security measures and reinforcing digital infrastructures against these persistent threats.*

Keywords*:* IoT Security, DDoS Attacks, Cyber Threats, Machine Learning , Network Security

I. INTRODUCTION

These days, almost every aspect of contemporary life is impacted by the "IoT" [1]. A diverse array of devices that comprise the IoT, each with a different technical background, leaves them open to potential security risks. Each entity has different security basics and qualities, thus it's become difficult to find a single solution that can safely solve every issue. Attackers may choose to target IoT devices due to insufficient security infrastructure. Furthermore, the Internet's service offering makes it possible to conduct banking and financial operations, communicate, engage in e-commerce, shop, make payments online, access healthcare, and get an education online [2]. The aforementioned services are particularly susceptible to cyber-attacks due to their extensive use. The most prevalent and deadly kind of cyber-attacks are DDoS attacks [3]. Numerous services are being interrupted.

Denial of service, or DoS, is an acronym describing what happens when a system delivers a malicious message to a server. When several hacked systems or computers launch DoS assaults against a single application, it's known as a DDoS attack. A deluge of packets from all corners of the globe is thereafter sent towards the designated network. DDoS attacks are becoming more frequent and sophisticated as a result of the spread of disruptive Internet technologies [4][5]. Cyber threats that might seriously affect a business's operations include ransom demands from attackers, data theft, and disruptions.

Responding quickly to DDoS assaults is the best way to prevent them. Cyber-attacks against internet-connected devices have become more appealing as a target due to the expanding use of the internet. As ML and DL [6][7] reveal their enormous potential in multiple areas, academics and industry are investigating the notion of using these technologies for DDoS detection. Traditional approaches are slower and less accurate when it comes to risk detection. Using an ML method, threats may be identified. DL may thus be a useful DDoS detection technique.

1.1 Contribution of Research

This work contributes to the field of cybers ecurity by implementing ML techniques for the classification and prediction of DDoS attacks. These study main contributions are:

- Implementation of ML models for DDoS attack detection and classification with the UNSW-NB15 dataset.
- Feature selection using Select K-Best method with the ANOVA F-test to identify relevant features.
- Data normalization using Min-Max Scaler to ensure consistent data scaling.
- Application of Cat Boost, ETC for robust prediction performance.
- Metrics for assessing the model's efficacy, including F1- score, recall, accuracy, and precision.

1.2 Organization of research work

The research is structured as follows for the sections that follow: In Section 2, the study's context is examined. Section 3 provides a full approach for this investigation. In Section 4,talk about the study's conclude the work and assessments. Findings from the research and recommendations for the future Section 5.

II. RELATED WORK

Machine learning/deep learning (ML/DL) has previously shown to be an effective method for identifying DDoS assaults. Some of the previous researchers work explained below:

Jiyad et al., (2024), presents a novel ensemble model that can identify DDoS attacks. The approach leverages ML algorithms such as LR, RF, DT, and XGBoost classifiers to detect and classify these malicious attacks effectively. In the research, use the potent explainable Artificial Intelligence (XAI) models SHAP and LIME. By utilizing SHAP and LIME's capabilities, improve the ML models' readability and transparency, giving us a better understanding of difficult predictions and model behavior. The evaluation results demonstrate that the XGBoost ensemble model outperforms other classifiers, obtaining a remarkable 97% accuracy rate and a remarkable 97% F-score. Accordingly, the precision and recall are 98% and 96%, respectively [8].

Al-Eryani, Hossny and Omara, (2024), focuses on providing a comparative study between recent ML algorithms that were tested using the CICDoS2019 dataset. This comparison aims to identify the best machine learning approach for DDoS detection. According to the findings of the comparative study, it is found that the Gradient Boosting (GB) and the XGBoost algorithms are extraordinarily accurate and correctly predicted the type of network traffic with 99.99% and 99.98% accuracy respectively, in addition to, a low false alarm rate of approximately 0.004 for GB[9].

Kaur, Sandhu and Bhandari, (2023), developed effective ML classifiers utilising attributes from the SDN dataset to identify DDoS assaults at the application layer. To narrow down the feature set of data, they have used ICA, PCA, and LDA. Furthermore, ML classifiers are developed using extracted characteristics, and DDoS attack prediction is carried out at the application layer. Out of 13, one feature was recovered using the LDA model, which provides the highest detection accuracy possible for the classifiers in use. Results are analysed by comparing the suggested work to earlier research. The study's result analysis using DT, RF, and SVC is accomplished up to 99.6%[10].

Patil et al., (2022), create a model based on ML to forecast DDoS flooding assaults. The DDoS flooding assaults that are to be expected encompass several kinds. These assaults were classified using ML models such as decision tree classifiers, MLP, KNN, and LR. A Jupiter notebook with the necessary Python libraries loaded was used for the implementation. KNN and DTC have shown almost identical performance, with the highest accuracy of 99.98%, in predicting TCP as well as ICMP flooding attacks out of these four classifiers. When it came to predicting UDP flooding attacks, the DTC performed a best, with an accuracy rate of 77.23 percent[11].

Cyber security is a critical topic in the field of internet security (Tufail, Batool and Sarwat, 2022). Cyber attacks affect many industries, with thousands occurring year. DDOS and FDIA are two of the most deadly cyberattacks. Two machine learning techniques, LR and SNN, were compared in this research in order to predict DDoS assaults. 99.85%

IJARSCT

International Journal of Advanced Research in Science, Communication and Technology

International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal

ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

accuracy was attained for SNN and 98.63% accuracy in logistic regression, respectively. In contrast to logistic regression, the analysis reveals that SNN required a significantly longer training period [12].

Despite significant advancements in ML approach for DDoS attack identification and classification, several gaps remain in the current research. While numerous studies have demonstrated high accuracy using various algorithms, Comprehensive comparisons across various datasets and attack types are lacking. This study, showcase impressive performance with XGBoost and Gradient Boosting, respectively, they do not address the performance consistency across different attack scenarios. Additionally, research focuses on specific attack types or datasets but lacks a holistic approach incorporating a wide range of attacks and feature reduction techniques. Furthermore, the computational efficiency and scalability of models are not thoroughly explored. Closing these shortcomings could improve DDoS detection systems' resilience and applicability. Table 1 present the related research on DDoS Attacks using ML and DL techniques provides a thorough summary of related work.

Table 1 Related Work on DDoS Attacks using ML and DL Techniques

References	Approaches	Dataset	Performance	Limitation
Jiyad et al. (2024)	LR, RF, DT, XGBoost + SHAP, LIME (XAI tools)	Custom dataset	XGBoost: Accuracy 97%, F- score: 97%, Precision: 98%,Recall: 96%	Limited to a specific dataset, lacks real-time implementation analysis
Al-Eryani, Hossny, and Omara (2024)	Gradient Boosting, XGBoost	CICDoS2019	GB Accuracy: 99.99%, XGBoost Accuracy: 99.98%	Focuses only on ML algorithms, no DL models explored
Kaur, Sandhu, and Bhandari (2023)	PCA, LDA, ICA with Decision Tree, Random Forest, SVM	SDN dataset	LDA Accuracy: 99.6% with ML classifiers	Limited to application- layer DDoS attacks, lacks DL exploration
Patil et al. (2022)	LR, KNN, MLP, DT	Custom dataset	KNN & Decision Tree: 99.98% (TCP/ICMP attacks), Decision Tree: 77.23% (UDP attacks)	Lower accuracy for UDP attack prediction (77.23%), only classical ML methods
Tufail, Batool, and Sarwat (2022)	Logistic Regression, Shallow Neural Network (SNN)	Custom dataset	SNN Accuracy: 99.85%, Logistic Regression: 98.63%	High training time for SNN, no other DL models evaluated

III. METHODOLOGY

There are Nemours stages and phases included in the strategy that has been presented. Machine learning methodologies and techniques are utilized in DDoS attack classification and prediction. For this project's implementation, the Python programming language was used. Implementation work additionally makes use of Python packages and libraries, including NumPy, seaborn, matplotlib, Pandas, Matplotlib, etc. The proposed methodology's first step is data collection. This research utilises the UNSW-NB15 datasets that is obtained from the Kaggle website. after data collection, conduct pre- processing to check the dataset's shape, remove missing or duplicate values, and perform label encoding on categorical columns. Then perform the feature selection task using select k-best methods with the ANOVA F-test. Next, normalize the data with the help of Min-max scaler methods. After that, the dataset is divided into 80% for training and 20% for testing. For classification, Cat Boost and Extra Tree classifiers are used to predict DDoS attacks. Next, determine the model's effectiveness using precision,f1- score, accuracy, and recall, as performance metrics. The flowchart in Figure 1 outlines the stages and subsequent steps of the suggested methodology.

3.1 Data Collection

For Classification and Prediction Techniques for DDoS Attacks data collection is a very initial step. in this study, collect the UNSW_NB15 dataset1 from publicly available sources. This dataset contains the following nine types of attacks: exploits worms, shellcode, DoS, backdoors, fizzers, and reconnaissance. To produce 49 characteristics with the class label, twelve algorithms are constructed in conjunction with the Argus and Bro-IDS tools. 2 million and 540,044 records in all are kept in four CSV files: UNSW-NB15_1.csv, UNSW-NB15_2.csv,UNSW-NB15_3.csv, and UNSW-NB15_4.csv.

3.2 Data Pre-processing

Reduced accuracy and prediction rate are the results of data preparation eliminating confusing data from the acquired dataset. It is necessary to exclude the possibility of human error as the cause of data loss prior to training the model. Datasets undergo further pre-processing after collection to eliminate duplicate or missing values. The datasets is then utilised for training the model after unnecessary values have been removed. Further pre-processing areas are defined in below:

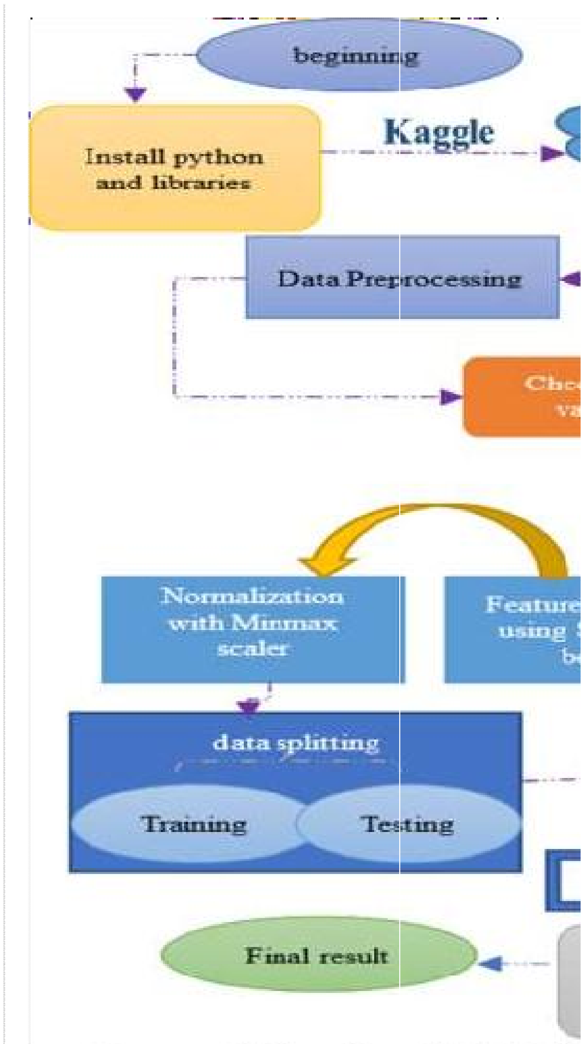


Fig. 1 Proposed Flowchart for DDoS Attacks Prediction

3.3 Label Encoding on the Categorical Column

Categorical variables are those that can take on a small, fixed range of values. Some examples of these factors include colour (red, blue, green), size (small, medium, big), and location (city, suburban, rural, etc.) [13]. Encoding categorical variables may be done in a number of ways. Label Encoding is one approach; it entails assigning a number value to

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

each separate category. For a colour characteristic that includes green, blue, and red categories, for example, the corresponding encoded values would be 0, 1, and 2, respectively. Keep in mind that this method may mislead the model if it unintentionally implies an ordinal connection among the numerical variables.

3.4 Feature Selection using Select k-Best with Anova f-Test

The first step is to partition the dataset according to the features and the variable of relevance [14]. After that, find the most significant features by using the SelectKBest technique when combined with the ANOVA F-test. Select the desired number of features to be preserved. To find the best features, the SelectKBest technique takes each feature's score relative to the target variable and uses that score to choose the top k features [15]. To enhance the model's functionality, this method focuses on the features that are most strongly related to the dependent variable.

3.5 Normalization with Minmax Scaler

Normalisation, or Min-Max scaling, is a commonly used method. To make values lie between 0 and 1, this approach adjusts and rescales the values [16]. The formula (1) is used to do the transition.

$$x' = x - \frac{x_{min}}{x_{max} - x_{min}} \tag{1}$$

In where x' stands for a normalized value, x' for an original value, and $xmax$ and $xmin$ for a maximum and lowest values of the corresponding feature.

3.6 Train-Test Split

A dataset's ability to be divided into training and testing portions is crucial for both model assessment and a deeper understanding of the properties of models. The ML model is fitted using a train dataset. However, the test dataset is utilized to evaluate a ML model. In this study, data have been used 80 percent for training and 20 percent for testing for better performance.

3.7 Classification Models

The proposed method includes machine-learning algorithms. This study uses Cat Boost, and Extra tree classifier for DDos attack prediction. Each classifier describes in below:

3.7.1 Extra Tree Classifier

The RF model served as the initial inspiration for the development of the Extra Tree classifier (ETC) technique, which was proposed by [17]. The ETC algorithm creates a set of unpruned judgements, or regression trees, in accordance with the traditional top-down methodology. The RF model uses bootstrapping and bagging, respectively, in two phases to achieve the regression. During the bootstrapping phase, a random training dataset sample is used to fuel the development of each individual tree, resulting in a collection of decision trees. After the DT nodes reach the ensemble, they are divided into groups using the two-step bagging phase. Many subsets of training data are chosen at random in the initial bagging stage. Making a choice is finished when the optimal subset and its value are selected.

The RF technique is made up of a series of decision trees, where the Gth prediction tree is presented by G(x, θr), and θ is a uniform independent distribution vector that is provided before the tree develops. By averaging each tree, equation (2) builds an ensemble of trees of G(x), therefore forming a forest.

$$(x, \theta_{1\ldots}\ \theta_{r}) = \frac{1}{\_}\sum_{=1}^{R} G(x,\ )_{r} \tag{2}$$

The ETR and RF systems differ from one another in two important ways. The ETR first separates nodes by randomly selecting a subset of all the cutting points. Secondly, to reduce bias, it cultivates the trees using all of the learning samples. The parameters k and nmin, which determine the minimum sample size needed to separate nodes, indicate the number of attributes that are randomly picked for each node in the ETR approach. The splitting procedure is controlled

ISSN 2581-9429 IJARSCT

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

by these variables. Also, k and nmin, respectively, dictate the intensity of the attribute selection and the average strength of output noise. The ETR model's accuracy is increased and overfitting is decreased by these two parameters [18][19].

3.7.2 Cat Boost Classifier

Cat Boost is a GBDT system that uses a less parameterised oblivious tree as its basic learner. It achieves good accuracy and supports categorical variables. Improves the algorithm's accuracy and applicability by training a sequence of learners sequentially using the boosting approach and then accumulating their results[20]. Concerning a training set of n samples, where can I get the labelled values and m-dimensional input features? After the training is complete, a powerful learner is created. The goal of the subsequent training is to choose a tree from the CART decision tree set T that minimises the expectation of the loss function. Our parameter calculation looks like this:

$$tk = argmi(y, F_{k-1}(x) + t(x)) \qquad (3)$$

Training samples and testing samples are not the same thing. The initial weak learner and the -th round of the training step size following iterations are used to create Model M, which is displayed in Equation (3.4). The loss function's negative gradient is applied in order to match the trained CART decision tree.

$$M = M_0 + \sum_{k=1} a_n t_k \qquad (4)$$

In comparison to previous boosting algorithms, Cat Boost improves upon the classic GBDT and introduces the following new features:

- The Cat Boost algorithm incorporates order boosting to counteract the training set's noise points [21];
- Cat Boost automatically converts categorical features to numerical features using the Ordered TS technique to enhance direct support for these features..;
- The introduction of categorical characteristics further enhances a feature dimension in Cat Boost; and
- Based on a completely symmetric tree, it applies same splitting criteria to each layer, leading to faster predictions and more stability [22].

IV. EXPERIMENT AND DISCUSSION

This work streamlines package management and distribution using the widely-used scientific computing programming language, Python. This system comes pre- installed with essential machine learning libraries such as Keras, Pandas, NumPy, Seaborn, Matplotlib, Scikit-learn, and TensorFlow, enabling efficient model development and data processing. The hardware setup for the pre-processing phase includes a system equipped with an Intel (R) Core (TM) i5- 12400F 4.4 GHz, , 2 Cores, and 4 Logical Processors, along with 16 GB of RAM and a 512 GB SSD. Additionally, for computationally intensive tasks, Google Research provides access to dedicated GPUs and TPUs, enhancing a performance of ML models used in this project.

4.1 Exploratory Data Analysis

This section of the research uses exploratory data analysis, or EDA, to look at the data closely. To facilitate understanding, this study employs a graphical representation of the data. To investigate the data and gather a synopsis of the most important findings, EDA is used. You may utilize its statistical insights and visualizations to help you find patterns or trends. The following data visualization graphs are provided in this section.

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

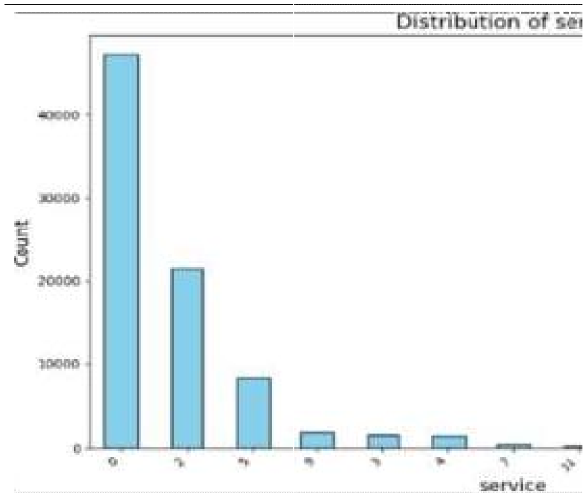


Fig. 2 Count Plot for Distribution of Service on UNSW_NB15 Data

The following Fig. 2 represents the Count plot for the Distribution of service on UNSW_NB15 data. Values on the"count" y-axis may go up to 40,000, while values on the "service" x-axis can go from 0 to 6. The tallest bar corresponds to service value “0,” indicating the highest count (well above 40,000)

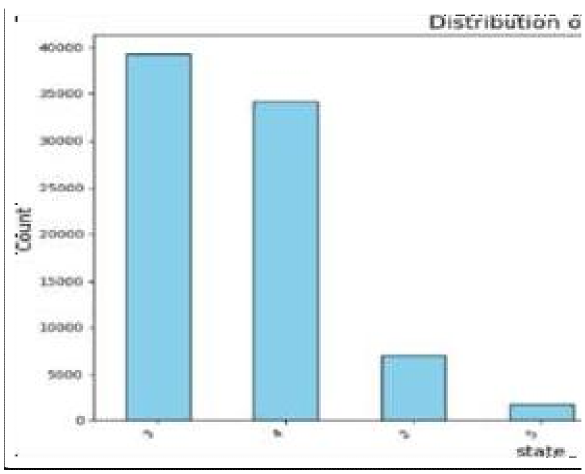


Fig. 3 Count plot for Distribution of state on UNSW_NB15 data

The distribution of seven network traffic states is shown in figure 3 by the count plot of the UNSW_NB15 dataset. The x-axis represents "state," and the y-axis indicates "COUNT." The first two states have significantly higher counts (around 40,000 and 35,000), while the remaining states range from 10,000 to 5,000, and the last state has a count of 0.

DOI: 10.48175/IJARSCT-30903

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

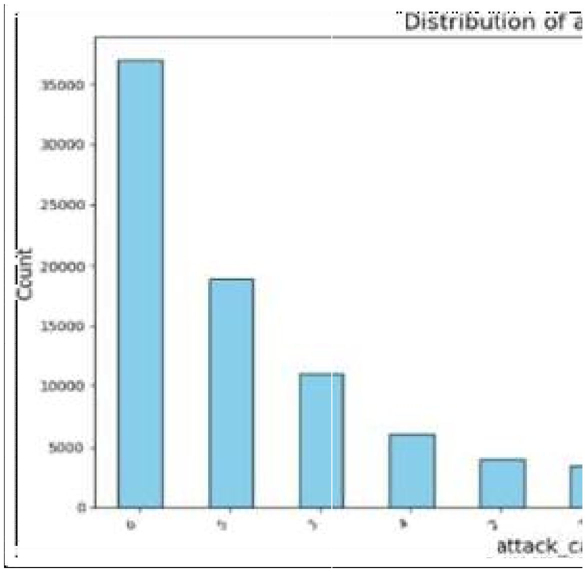


Fig. 4 Count Plot for Distribution of Attack_cat on UNSW_NB15 Data

The bar graph Distribution of attack cat on UNSW_NB15 data displays in figure 4 the count of 9 different categories of attacks on the x-axis and their respective counts on the y- axis. The first bar is significantly taller, indicating a higher frequency for that attack category. Although the exact labels for the categories are not visible, the graph effectively shows the overall distribution of cyber-attacks within the dataset.

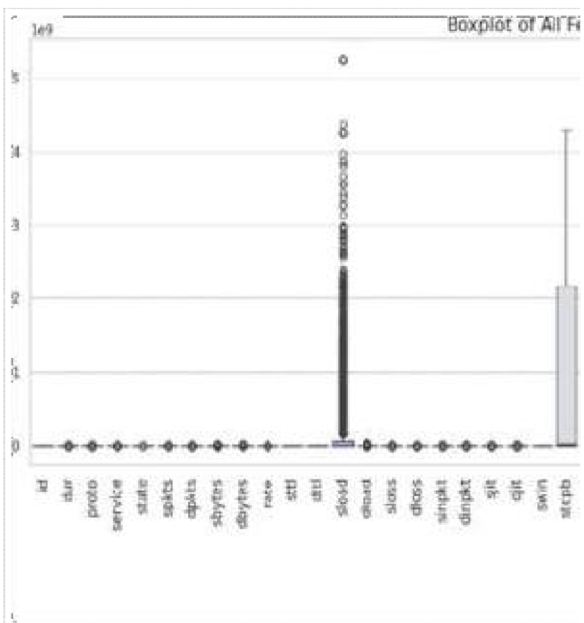


Fig. 5 Box Plot for Features in UNSW_NB15 Data

The box plot for features in the UNSW_NB15 dataset displays in figure 5, various features on the x- axis, such as 'dur', 'spkts', 'dpkts', and 'sbytes', while the y-axis, scaled logarithmically, shows the values of these features. Each box represents the distribution of a feature, indicating the median(line inside the box), quartiles (box edges), and potential outliers (dots beyond the whiskers). This visualization facilitates quick comparison of central tendency, variability, and outliers across different features

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

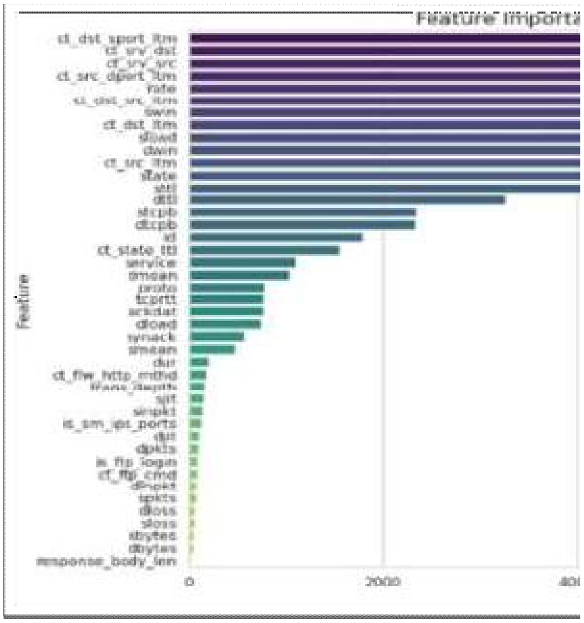


Fig. 6 Feature Importance Score Graph

Figure 6 display the Feature important score graph generated by SelectKBest. The y-axis represents various features (such as „ct_dst_sport_ltm", „ct_src_dport_ltm", etc.). The x-axis shows the importance scores, ranging from 0 to 8000. Each feature has a corresponding bar, with its length indicating its importance score.

4.2 Evaluation Parameter

Model performance may be better understood with the use of evaluation metrics. The ability of evaluation metrics to differentiate between different model outputs is a key feature. In general, the values used to compute these measures are obtained from the confusion matrix (see figure 7 below), which displays the correctness of the model in a very intuitive way. This matrix is N X N, where N is the projected number of classes.

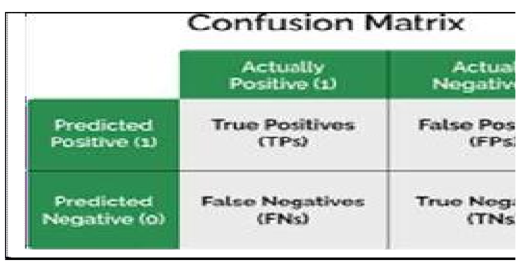


Fig. 7 Representation of Confusion Matrix

The four-class classification system divides instances (examples) into four separate groups. Class A, Class B, Class C, and Class D are the four groups that comprise the whole. Positive (1) and negative (0) stand for the expected values, whereas true (1) and false (0) indicate the actual values. The confusion matrix expressions TP, TN, FP, and FN are used to derive estimates of the possible classification models.

Accuracy

The percentage of correct forecasts compared to the total number of predicts is known as accuracy. Equation (5) was used to calculate accuracy.

$$Accuracy - \frac{TN+TP}{TP+TN+FP+FN} \quad (5)$$

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429 Volume 6, Issue 3, January 2026 Impact Factor: 7.67

Recall

Equation (6) provides recall, which can be defined as the ratio of positively classified samples to all samples in the real class (including both TP and FN samples).

$$Recall - \frac{TP}{TP+FN} \quad (6).$$

Precision

The precision measures how many positive samples (FP and TP combined) were properly detected out of all the positive samples. The focus is mostly on how well the model detects positive samples. There is a formula that follows (7)

$$Precision - \frac{TP}{TP+FP} \quad (7)$$

F1 score

The F1 score is primarily composed of two components: precision and recall. Both FP and FN classified samples are taken into consideration by the F1-score. Having an equal number of FP and FN samples will improve finding accuracy. The following formula (8).

$$F1\,Score - \frac{2{*}Precisio \times Recall}{Precisio+Recall} \quad (8)$$

4.3 Results Analysis

The proposed model extra tree and Cat Boost model performance across performance parameters is provided in this section. The following table 2 provides the model performance which shows both models achieve the highest performance across performance parameters. The ETC model achieve 90.27% accuracy and Cat boost achieved 90.78% accuracy.

Table 2 Proposed model Performance on the UNSW_NB15 Dataset

Performance metric	ETC	Cat Boost
Accuracy	90.27	90.78
Precision	89.86	90.58
Recall	90.27	90.78
F1-score	89.89	90.37

Bar Graph for proposed model performance shows in figure 8. When comparing the performance metrics between ETC and Cat Boost, both models demonstrate strong capabilities across accuracy, precision, recall, and F1-score. Cat Boost slightly outperforms ETC in accuracy (90.78% vs. 90.27%) and precision (90.58% vs. 89.86%), showing a slight edge in correctly predicting positive instances and minimizing false positives. Recall scores are identical for both models at 90.27%, indicating they equally capture true positive instances. F1-scores also favor Cat Boost slightly, achieving 90.37% compared to ETC's 89.89%, reflecting a better balance between precision and recall. Overall, while both models perform exceptionally well, Cat Boost demonstrates slightly superior performance in accuracy and F1-score, making it a favorable choice for tasks requiring robust predictive performance.

ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

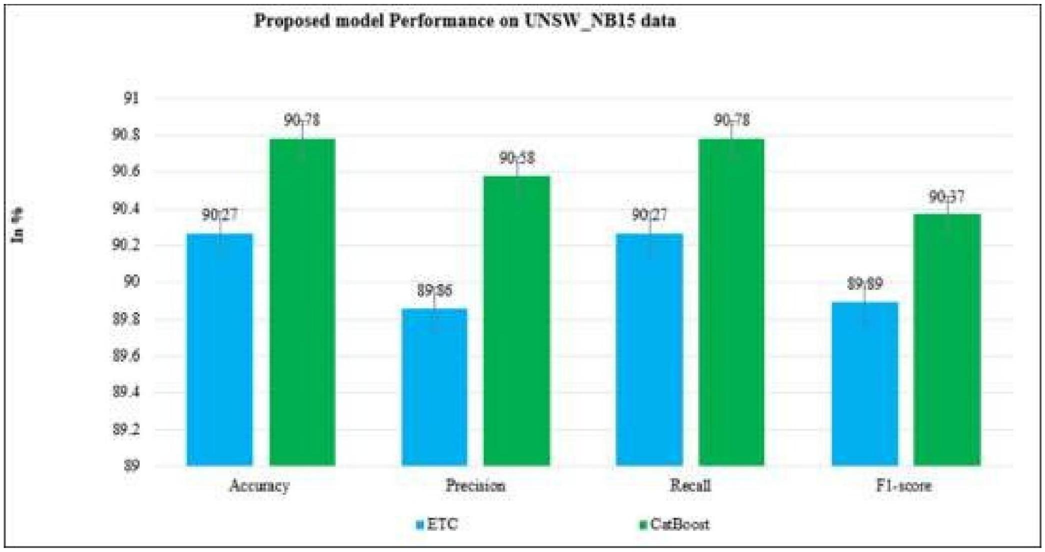
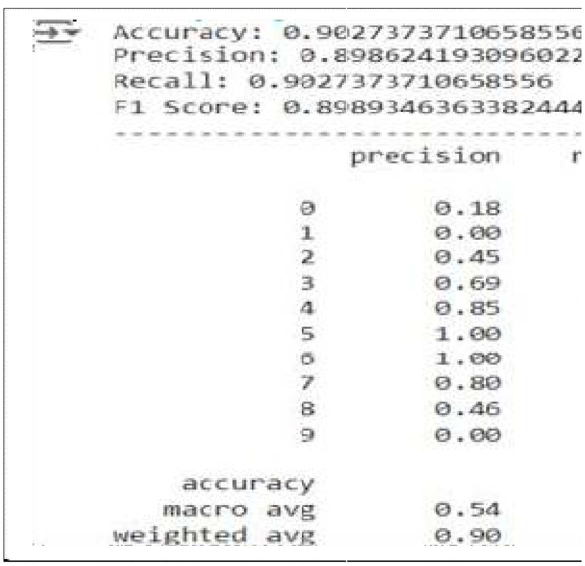


Fig. 8 Bar Graph for proposed model performance



```
Accuracy: 0.9027373710658556
Precision: 0.898624193096022
Recall: 0.9027373710658556
F1 Score: 0.8989346363382444
-----------------------------
              precision    r

           0       0.18
           1       0.00
           2       0.45
           3       0.69
           4       0.85
           5       1.00
           6       1.00
           7       0.80
           8       0.46
           9       0.00

    accuracy
   macro avg       0.54
weighted avg       0.90
```

Fig. 9 Classification Report of Extra Tree Classifier

DOI: 10.48175/IJARSCT-30903

IJARSCT
International Journal of Advanced Research in Science, Communication and Technology
International Open-Access, Double-Blind, Peer-Reviewed, Refereed, Multidisciplinary Online Journal
ISSN: 2581-9429
Volume 6, Issue 3, January 2026
Impact Factor: 7.67

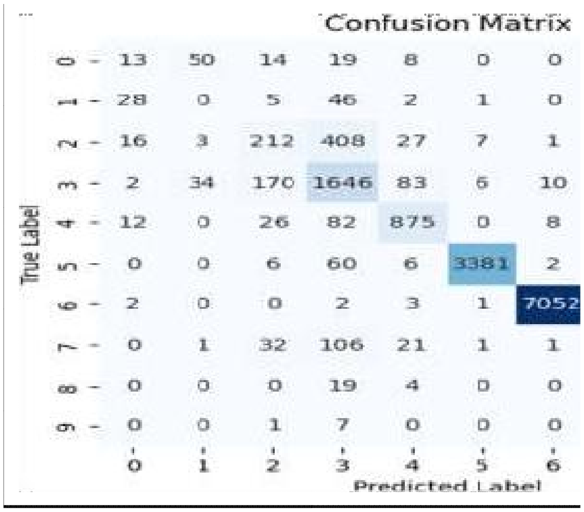
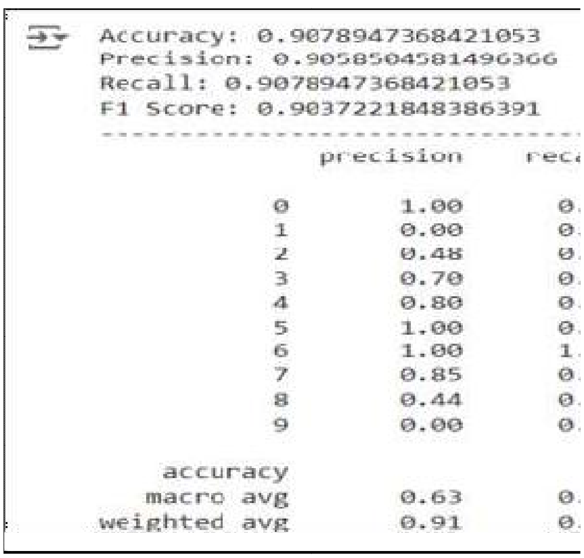


Fig. 10 Confusion matrix for Extra tree classifier

The confusion matrix of an ETC is demostrate in Fig. 10, where the real class labels (0–9) are shown on the y-axis, and the predicted class labels are represented on the x-axis. More predictions for a true- predicted label pair are represented by deeper hues in each cell. Diagonal cells stand for each class's accurate predictions, also known as true positives.



```
Accuracy: 0.9078947368421053
Precision: 0.9058504581496366
Recall: 0.9078947368421053
F1 Score: 0.9037221848386391
------------------------------
              precision    rec
           0       1.00     0.
           1       0.00     0.
           2       0.48     0.
           3       0.70     0.
           4       0.80     0.
           5       1.00     0.
           6       1.00     1.
           7       0.85     0.
           8       0.44     0.
           9       0.00     0.

    accuracy
   macro avg       0.63     0.
weighted avg       0.91     0.
```

Fig. 11 Classification Report of CatBoost Classifier

Figure 11 illustrates the Cat Boost classifier's classification report, which includes 10 classes. The classifier's accuracy is 90.79%, showing a good match among model predictions and labels. The Precision of Cat Boost classifier is 90.58, recall is 90.78, and f1-score is 90.37. The model displays varied performance across different classes: it excels in precision for classes 0, 5, and 6 but struggles with recall in classes 0, 8, 1, and 9. Classes 3, 4, and 7 show moderate to good performance with balanced precision and recall. The overall accuracy of 0.91 with 15124 support value.

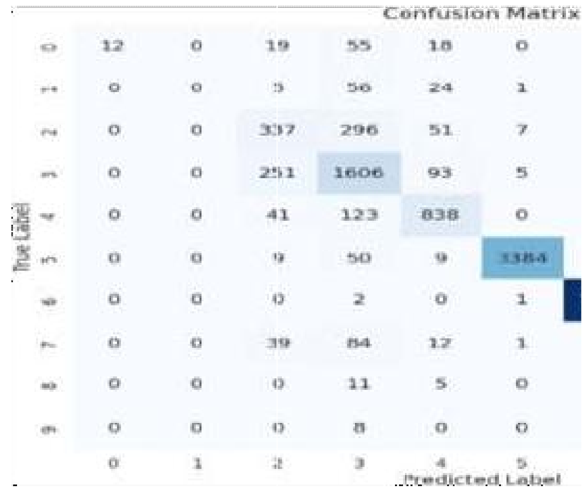


Fig. 12 Confusion Matrix for CatBoost Classifier

Figure 12 displays the confusion matrix for the Cat Boost classifier. In this figure, The predicted labels are shown on the x-axis, and the actual labels are shown on the y-axis. Both axes range from 0 to 9. Correct predictions are along the diagonal, with darker blue indicating higher counts, like 7058 for class 6. Off- diagonal cells show misclassifications, such as 55 instances where true label 0 was predicted as 1. This matrix helps identify correct classifications and common confusions, guiding model improvements.

4.4 Comparative Study

The Comparison of Base and proposed model performance across performance parameters is provided in this section. The model performance comparison in Table 3 below demonstrates how well the suggested model performs in contrast to basic models.

Table 3 Comparison of base and Propose model Performance on UNSW_NB15 Dataset

Performance Metric	Propose Models		Base Models	
	ETC	Cat Boost	RF	XGBoost
Accuracy	90.27	90.78	88.94	89.95
Precision	89.86	90.58	89.03	90.89
Recall	90.27	90.78	88.94	89.95
F1-score	89.89	90.37	88.96	89.67

Comparing the performance metrics of proposed ensemble models (ETC and Cat Boost) against base models (RF and XGBoost) reveals consistently high performance across performance metrics shows in table 3. The figure show higher accuracy and precision, with Cat Boost slightly ahead in precision at 90.58%. Recall scores are equally strong across all models, matching accuracy levels closely. F1- scores show Cat Boost leading marginally at 90.37%, indicating balanced performance in precision and recall. Overall, the ensemble models of ETC and Cat Boost demonstrate robustness and reliability, making them effective choices for scenarios requiring high predictive accuracy and comprehensive model performance.

V. CONCLUSION AND FUTURE SCOPE

The emergence of applications for intelligent buildings raises the possibility of cybersecurity risks for people, companies, and the technology they use. The study emphasises how crucial it is to use machine learning methods in cybersecurity, particularly when accuracy and speed are critical. While research based on ML provide encouraging results, this study shows that deep learning is not the only approach that works. Models that are straightforward, understandable, and practical may be used to counter DDoS assaults. This study aimed to advance the classification as well as prediction of DDoS attacks by employing sophisticated machine learning methodologies on the UNSW-NB15 datasets. This work showed how well several ML methods, including Extra Tree and Cat Boost, can be used to the detection and categorisation of DDoS assaults. Specifically, Cat Boost delivered an accuracy90.78%, precision90.58%, recall90.78%, and an F1-score90.37%, Both Cat Boost and Extra Tree classifiers outperformed the base models across all metrics, including F1-score, recall, accuracy, and precision. This comparative edge indicates that the proposed models not only provide superior detection and prediction of DDoS attacks but also enhance overall system robustness. The outcomes demonstrate the proposed methodology's efficacy and dependability while emphasising its potential to greatly enhance intrusion detection systems' capacity to recognise and address DDoS threats.

Conflict of Interest: None

REFERENCES

[1] S. Kumar, P. Tiwari, and M. Zymbler, “Internet of Things is a revolutionary approach for future technology enhancement: a review,” J. Big Data, 2019, doi: 10.1186/s40537-019-0268-2.

[2] M. Snehi and A. Bhandari, “Vulnerability retrospection of security solutions for software-defined Cyber- Physical System against DDoS and IoT-DDoS attacks,” Computer Science Review. 2021. doi: 10.1016/j.cosrev.2021.100371.

[3] R. K. C. Chang, “Defending against flooding-based distributed denial-of-service attacks: A tutorial,” IEEE Commun. Mag., 2002, doi: 10.1109/MCOM.2002.1039856.

[4] B. Patel, V. K. Yarlagadda, N. Dhameliya, K. Mullangi, and S. C. R. Vennapusa, “Advancements in 5G Technology: Enhancing Connectivity and Performance in Communication Engineering,” Eng. Int., vol. 10, no. 2, pp. 117–130, 2022, doi: 10.18034/ei.v10i2.715.

[5] R. K. Gupta, K. K. Almuzaini, R. K. Pateriya, K. Shah,P. K. Shukla, and R. Akwafo, “An Improved Secure Key Generation Using Enhanced Identity-Based Encryption for Cloud Computing in Large- Scale 5G,” Wirel. Commun. Mob. Comput., 2022, doi: 10.1155/2022/7291250.

[6] V. Rohilla, S. Chakraborty, and M. Kaur, “An Empirical Framework for Recommendation-based Location Services Using Deep Learning,” Eng. Technol. Appl. Sci. Res., 2022, doi: 10.48084/etasr.5126.

[7] P. Khuphiran, P. Leelaprute, P. Uthayopas, K. Ichikawa, and W. Watanakeesuntorn, “Performance comparison of machine learning models for DDoS attacks detection,” in 2018 22nd International Computer Science and Engineering Conference, ICSEC 2018, 2018. doi: 10.1109/ICSEC.2018.8712757.

[8] Z. M. Jiyad, A. Al Maruf, M. M. Haque, M. Sen Gupta,A. Ahad, and Z. Aung, “DDoS Attack Classification Leveraging Data Balancing and Hyperparameter Tuning Approach Using Ensemble Machine Learning with XAI,” in 2024 Third International Conference on Power, Control and Computing Technologies (ICPC2T), 2024, pp. 569–575. doi: 10.1109/ICPC2T60072.2024.10475035.

[9] A. M. Al-Eryani, E. Hossny, and F. A. Omara, “Efficient Machine Learning Algorithms for DDoS Attack Detection,” in 2024 6th International Conference on Computing and Informatics (ICCI), 2024, pp. 174–181. doi:10.1109/ICCI61671.2024.10485168.

[10] S. Kaur, A. K. Sandhu, and A. Bhandari, “Feature Extraction and Classification of Application Layer DDoS Attacks using Machine Learning Models,” in 2023 International Conference on Communication, Security and Artificial Intelligence, ICCSAI 2023, 2023. doi: 10.1109/ICCSAI59793.2023.10421652

[11] P. S. Patil, S. L. Deshpande, G. S. Hukkeri, R. H. Goudar, and P. Siddarkar, “Prediction of DDoS Flooding Attack using Machine Learning Models,” in Proceedings of the 3rd International Conference on Smart Technologies in Computing, Electrical and Electronics, ICSTCEE 2022, 2022. doi: 10.1109/ICSTCEE56972.2022.10100083.

[12] S. Tufail, S. Batool, and A. I. Sarwat, “A Comparative Study Of Binary Class Logistic Regression and Shallow Neural Network For DDoS Attack Prediction,” in Conference Proceedings - IEEE SOUTHEASTCON, 2022. doi: 10.1109/SoutheastCon48659.2022.9764108.

[13] W. Yustanti, N. Iriawan, and Irhamah, “Categorical encoder based performance comparison in preprocessing imbalanced multiclass classification,” Indones. J. Electr. Eng. Comput. Sci., 2023, doi: 10.11591/ijeecs.v31.i3.pp1705-1715.

[14] V. Rohilla, S. Chakraborty, and R. Kumar, "Deep learning based feature extraction and a bidirectional hybrid optimized model for location based advertising," Multimed. Tools Appl., vol. 81, no. 11,pp. 16067–16095, May 2022, doi: 10.1007/s11042-022-12457-3.

[15] R. C. Chen, C. Dewi, S. W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," J. Big Data, 2020, doi: 10.1186/s40537-020-00327-4.

[16] A. Bhandari, "Feature Engineering: Scaling, Normalization and Standardization," Analytics Vidhya.

[17] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," Mach. Learn., vol. 63, no. 1, pp. 3– 42, 2006, doi: 10.1007/s10994-006-6226-1.

[18] V. John, Z. Liu, C. Guo, S. Mita, and K. Kidono, "Real- time lane estimation Using Deep features and extra trees regression," in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2016. doi: 10.1007/978-3-319-29451- 3_57.

[19] G. Mishra, D. Sehgal, and J. K. Valadi, "Quantitative Structure Activity Relationship study of the Anti- Hepatitis Peptides employing Random Forest and Extra Tree regressors," Bioinformation, 2017, doi: 10.6026/97320630013060.

[20] A. V. Dorogush, V. Ershov, and A. Gulin, "CatBoost: gradient boosting with categorical features support,"pp. 1–7, 2018.

[21] L. Prokhorenkova, G. Gusev, A. Vorobev, A. V. Dorogush, and A. Gulin, "Catboost: Unbiased boosting with categorical features," in Advances in Neural Information Processing Systems, 2018.

[22] H. Liu, L. Guo, H. Li, W. Zhang, and X. Bai, "Matching Areal Entities with CatBoost Ensemble Method," J. Geo-Information Sci., 2022, doi: 10.12082/dqxxkx.2022.220050.