International Journal of Engineering Sciences & Emerging Technologies, Dec. 2024
ISSN: 22316604 Volume 11, Issue 4, pp: 722-729 ©IJESET

A LIGHTWEIGHT MACHINE LEARNING BASED INTRUSION DETECTION SYSTEM FOR EDGE COMPUTING

Vipin Kumar[1], Vivek Kumar[1] and Amit Kumar[1,2]
[1]Department of Computer Science,
Gurukula Kangri (Deemed to be University), Haridwar, India
19523003@gkv.ac.in
vivekdcg@gkv.ac.in
[2] *Quantum University, Roorkee, India*
19523002@gkv.ac.in

ABSTRACT

Edge computing (EC) is a decentralized computing environment where data is processed near the end user, leading to lower latency and reduced bandwidth usage. Privacy and security concerns persist in spite of the wide range of uses for EC. EC poses a heightened vulnerability to network security breaches due to the transmission of data over several networks and its processing on various devices. There are several safe solutions available for this, including a firewall, robust authentication techniques, and an intrusion detection system (IDS). IDS are employed to identify zero-day attacks or detect previously undisclosed cyber-attacks in edge networks. This paper presents an IDS that utilizes ML techniques and is trained on the UNSW-NB 15 dataset. The IDS model being suggested utilizes a Filter-Based feature selection method and classification algorithm to detect and classify harmful network events accurately. The feature selection process is conducted using the relevance score approach, as well as the Decision Tree, Extra Tree, Random Forest, and Extended Gradient Boosting classification algorithms. The findings indicate that the classification model, employing fewer features, achieves enhanced accuracy owing to increased detection rate and reduced false positive rate. The efficacy of the suggested IDS paradigm for EC is demonstrated since it ensures a secure network environment for users.

KEYWORDS

Edge Computing, Network Security, Feature Selection, ML

1. INTRODUCTION

A computing paradigm describes the basic ideas behind how computers handle problems and data manipulation. It is a way of thinking about and studying computers from a cognitive perspective. Cloud, edge, IoT, and quantum computing are some of the newer paradigms in computing that have evolved throughout the years. Nowadays, practicality is of the utmost significance. With cloud computing, businesses may reduce costs, simplify implementation, and increase the scalability of their IT resources to suit their computing needs. Computer services are also available, flexible, and affordable [1]. The lack of oversight over the shared infrastructure and data makes cloud computing susceptible to security breaches, which is its main drawback. An answer to this issue is the Internet of Things (IoT) and Edge Computing (EC), which allow data processing to happen near the data source. As a result, latency is decreased, and efficiency is enhanced [2]. Microcloud, mobile edge, and fog computing are all components of edge infrastructures. Many different sectors can benefit from edge applications, such as healthcare, retail, finance, smart cities, and the industrial and automotive IoT [3]. Unauthorized users, data breaches, and cyberattacks are all possibilities with EC. Due to the dispersed nature of data storage and processing at the edge, which increases the likelihood of data breaches, EC presents serious threats to network security and privacy [4]. When it comes to EC, there are a number of ways to handle network security and data privacy concerns. Some of these tactics include encryption methods, intrusion detection systems, and firewalls [5][6]. Put in place Intrusion Detection

Systems (IDS) to forestall cyberattacks, data breaches, and unauthorized access. The objective of an Intrusion Detection System (IDS) is to monitor data transfer across a network and alert the administrator to any suspicious or malicious activity. Unauthorized access and cyber-attacks can be efficiently reduced with this strategy. Data security and privacy issues in anomaly-based intrusion detection system models and signature-based intrusion detection system EC are best addressed via AIDS. The reason behind this is that AIDS can identify incursions by comparing them to a normal baseline. Also, it may spot unusual behaviour and use that information to find new dangers. These days, computers can learn from data, see trends, and make decisions with little to no human input, thanks to Machine Learning (ML). With ML, EC and Intrusion Detection Systems (IDS) can analyze data in real time and spot trends that could indicate malicious activity [7].[8]. The EC systems improve overall operational efficacy and help to reduce security risks. In order to train ML models more effectively, feature engineering extracts the optimal number of features from a dataset. To find the best and most effective condensed collection of features for the ML model, feature selection (FS) procedures like filters, embedding methods, and recursive feature reduction are used [9]. The UNSW-NB15 dataset is a great choice for intrusion detection system research because of its large volume and wide variety of attack categories [10]. This is a collection of labelled network traffic captures. Anomaly detection systems that rely on networks can be trained, validated, and evaluated using these captures. Using a meta-classification method made possible by stacking generalization, Smitha Rajagopal et al. provide an ensemble model. In both simulated and actual network settings, this model is used with the UNSW NB-15 and UGR'16 datasets. For the real-time dataset, the results show an impressive 97% accuracy [11]. A. Thakkar and R. Lohia enhanced network security by utilizing FS methods to identify pertinent features, which were then classified using machine learning algorithms. The NSL-KDD dataset is used to assess the outcomes [12]. Kasongo and Sun applied XGBoost to reduce features in the UNSW-NB15 dataset. They utilized various ML approaches, such as DT (decision trees), artificial neural networks (ANN), logistic regression, k-nearest neighbors (KNN), and SVM (support vector machines). The results show that the DT test's accuracy goes up from 88.13% to 90.85% when XGBoost methods are used for binary classification [13]. Saif S. Karim and colleagues have proposed a new approach to improving performance in FS using GTO-BSA. GTO-BSA outperformed state-of-the-art methods in terms of convergence and solution quality when tested on four IoT-IDS datasets [14].

Here is the outline of the document: Part II describes the dataset in depth and describes the procedure followed to choose the features. Section III introduces feature reduction to its optimal form. Section IV details the experimental setup and the outcomes. Section V provides a thorough literature analysis of studies that have used the UNSW-NB15 dataset for attack classification. Section VI concludes the paper.

2. PROPOSED METHOD

Figure 1 demonstrates the foundational structure of the suggested intrusion detection system (IDS) model, which utilizes machine learning (ML). The methodology suggested in this paper utilizes ML methods to detect network intrusions and differentiate normal signals from aberrant signals using the UNSW-NB15 benchmark dataset.

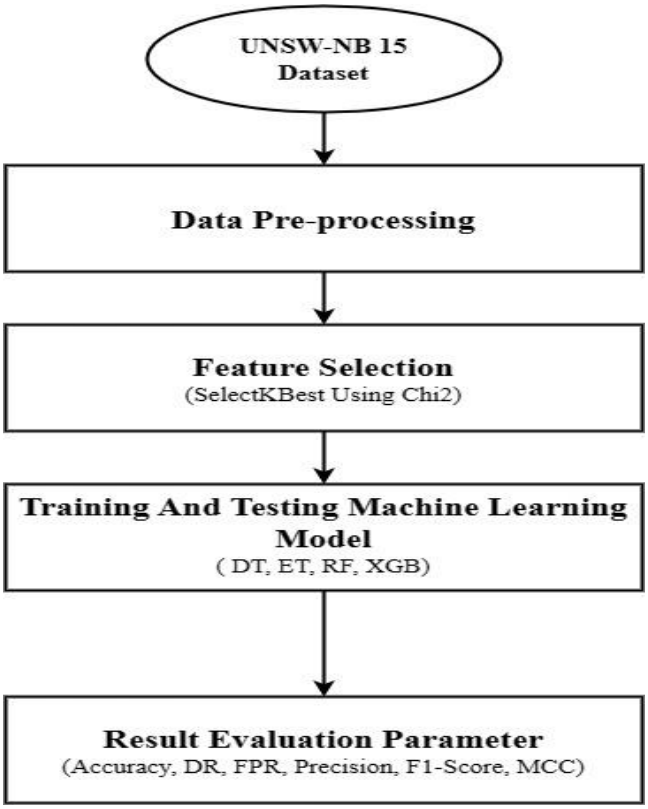

Figure1. ML-based proposed methodology

2.1. UNSW-NB15 Dataset

The UNSW-NB15 dataset, a comprehensive collection of real network traffic data, is of great significance in intrusion detection research. It encompasses a diverse range of attacks and common behaviors, serving as an ideal resource for academics to analyze attack patterns and evaluate the effectiveness of Intrusion Detection Systems (IDSs). The dataset provides substantial statistical data on each attack, enabling researchers to examine and compare different intrusion detection algorithms closely.

Developed by UNSW in 2015, the UNSW-NB15 dataset is meticulously tailored for detecting network intrusions. It consists of network traffic flows painstakingly categorized as either malicious or normal. The dataset is further divided into two separate sets: one for training and one for testing. The training dataset contains 823,430 entries, with 19,096 being assault records, while the test dataset contains 175,341 records, with 8,341 being attack records. Overall, the dataset features nine unique attack scenarios, 49 attributes, and a total of 2,540,829 network connections, making it a reliable resource for training and evaluating network intrusion detection systems.

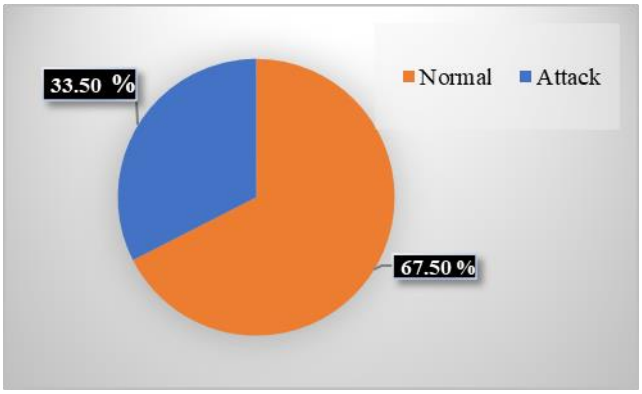

Figure 2. Labels in UNSW-NB 15 dataset

2.2 Data Preprocessing

The preprocessing stage serves to diminish the intricacy of the data, enhance accuracy, and prepare it for ML models, hence facilitating analysis [15-18]. Additionally, standardizing the data and addressing outliers or missing values aids in enhancing the accuracy of the model. The preprocessing stage of the UNSW-NB15 dataset primarily involves four sub-steps, as outlined below:

1. Elimination of superfluous attributes: Exclude any attributes that are not pertinent to the categorization process.
2. Standardization of numerical characteristics: Standardize numerical features to ensure they fall within the range of 0 and 1.
3. Categorical feature one-hot encoding: Transform categorical characteristics into a collection of binary variables to enable their utilization in ML techniques.
4. Feature Selection: Identify and choose the most significant features to incorporate into the ML model.

2.2 Selection of Optimal Features

FS, a crucial technique in machine learning, helps us identify the most important features that significantly impact prediction accuracy. By reducing the number of features, we enhance generalization, shorten training time, and simplify model complexity. This process, by eliminating unnecessary or redundant elements, improves model accuracy. What's fascinating is the diverse range of FS methods in machine learning, including filter methods, wrapper methods, embedding methods, and hybrid approaches, each with its unique approach and benefits. Filter methods analyze the relationship between each feature and the dependent variable, while wrapper approaches measure the effectiveness of a chosen set of features. Embedded techniques consider FS as an essential component of the model construction process, while hybrid methods integrate elements from the other three methods. Filter-based selection methods in machine learning are used to preprocess the data and discover a subset of features that are highly relevant to the topic being addressed. These methods utilize statistical measurements such as correlation, information gain, chi-squared, or other scores to prioritize the features and select only the most ideal ones. Afterwards, the selected attributes are used to build a machine learning model that demonstrates improved precision and superior ability to apply to unknown data. Machine learning (ML) utilizes filter-based selection methods to select a subset of features from a larger set of features [16-19]. These particular qualities are then employed in the machine learning model to produce forecasts and make well-informed decisions. Filter-based selection methods use statistical measures to assess the importance of each feature in the dataset. Features demonstrating a strong association with the output variable are selected, while the rest are discarded. This process streamlines the model and enhances its efficiency.

For this study, we utilized the SelectkBest (SKB) strategy, specifically the chi-square technique, which is a filter-based FS method, to accomplish our objective. The Chi-square technique measures the degree of correlation between two variables and is used to identify the most important aspects. The usefulness of this method is based on its capacity to identify non-linear relationships and aid in

the identification of significant features. We selected 28 attributes from SKB using Chi2 FST out of the total of 42 characteristics.

2.3 Training and Testing

To conduct an initial analysis of the dataset, we utilized the techniques in Colab Jupyter notebooks and employed the Python programming language, which is commonly used for constructing ML models. The experiment was conducted on a RedmiBook 15 Pro series laptop equipped with a 64-bit Intel(R) Core(TM) i5 processor running at 3.11 GHz and 8 GB of memory. The dataset was split, with 80% allocated for training and the remaining 20% reserved for testing.

2.4 Supervised ML Classifiers

Supervised ML classifiers utilize labelled training data to acquire knowledge and generate predictions for unseen data. Supervised learning algorithms utilize labelled data to acquire knowledge of the relationship between the inputs and outputs, enabling them to apply this knowledge to new, unseen data. Supervised learning algorithms are employed for many purposes, such as classification, regression, and forecasting. The following supervised ML classifiers are considered most favourable for the UNSW NB15 dataset:

1. Random Forest Classifier (RF): The Random Forest (RF) is a popular supervised machine learning classifier renowned for its ability to manage large datasets and high-dimensional data while producing accurate results. It is also resilient to outliers, has minimal bias, and is easy to implement.
2. A DT is a type of supervised ML algorithm that classifies data by asking questions and making judgments based on the replies. It is widely used in several applications, including image identification, NLP (natural language processing), and medical diagnosis.
3. The Extra Tree Classifier is a more sophisticated variant of the DT algorithm specifically designed to mitigate the problem of overfitting. The system employs a blend of methodologies, such as bagging, boosting, and random sampling, to attain superior precision and minimize variability.
4. The XGB Classifier is an ML method that utilizes gradient boosting for classification tasks. It is meant to efficiently and accurately process big datasets and generate highly precise predictions.

2.5 Performance Metrics

In our experiment, the performance metrics for evaluating the attack classification using a machine learning classifier are detailed below.

1. Accuracy:

$$A=(Tp+Tn) / (Tp+Tn+Fp+Fn) \quad \ldots(1)$$

2. Detection Rate (Recall):

$$DR =Tp / (Tp+Fn) \quad \ldots(2)$$

3. False Positive Rate (FPR):

$$FPR =Fp/(Fp+Tn) \quad \ldots(3)$$

4. Precision (PR):

$$PR = Tp / (Tp + Fp) \quad \ldots(4)$$

5. $F1\_Score = 2*Pr*Recall / (Pr + Recall)$ ...(5)

2.5 **Results and Analysis**

In the initial stage of UNSW-NB15, the dataset is properly preprocessed. The second phase employs the Chi2 method to select the most suitable features from the preprocessed data or to decrease the features. Classification models are then applied to various feature sets, including sets with 7, 14, 21, 28, 35, and all features. The SelectKBest FS method, which is based on the Chi2 technique, is used for this purpose.

TABLE 1. Different feature sets using importance score

FST	'k'		DT	ET	RF	XGB
SkB using Chi2 Tech.	7	DR	95.12	95.15	95.22	96.79
		FPR	2.27	2.26	2.28	2.90
	14	DR	96.28	96.33	96.52	95.93
		FPR	2.29	2.29	2.34	2.28
	21	DR	93.56	96.42	97.01	96.15
		FPR	2.07	1.96	2.00	2.33
	28	DR	95.86	96.72	97.17	96.00
		FPR	1.45	1.23	1.10	1.66
	35	DR	95.65	96.85	97.14	96.02
		FPR	1.52	1.31	1.28	1.63
	Full	DR	96.10	96.89	97.08	95.90
		FPR	3.90	3.11	1.15	1.62

The selection of the feature set is determined by the significance score assigned to each feature. The main goal of this research was to attain a high rate of detection and a low rate of false positive results on feature sets. In order to determine the minimum number of optimal features that can be selected for the model, this paper utilizes four ML classifiers: RF, DT, ET, and KNN. Each of the four classifiers is applied to all feature sets in the dataset, and the corresponding experimental results are displayed in Table 1. This paper compared ML classifiers based on precision, recall, F1 score, and false positive rate (FPR), as shown in Figure 3. According to the results from the UNSW-NB 15 dataset, the RF classifier has the maximum accuracy compared to all other classifiers, as depicted in Figure 3. Nevertheless, alternative classifiers also show strong performance.

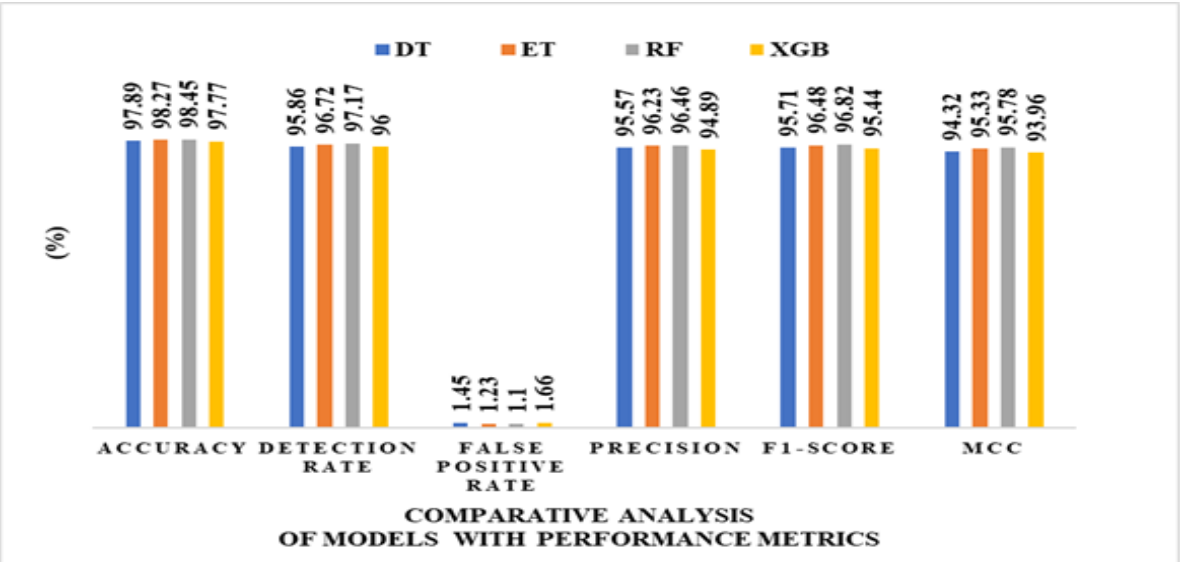

Figure 3. Comparison of ML models

3. CONCLUSIONS

This study presents a novel ML-based IDS. The experimental approach involves applying innovative FS techniques to extract optimal attributes from the UNSW-NB 15 dataset. Utilizing RF, DT, Extra Tree (ET), and XGBoost (XGB) classifiers, the research categorizes attacks on the benchmark dataset. The proposed methodology evaluates the effectiveness of these classifiers by eliminating unnecessary features. Our findings highlight that the RF classifier exhibits superior accuracy compared to the others. The experiments demonstrate that RF achieves high detection accuracy while maintaining a low false positive rate, aligning with the primary objective of this study. This underscores RF's effectiveness as an efficient Intrusion Detection System (IDS) with its minimal false negatives and robust detection of false positives. Future work will focus on employing advanced feature engineering strategies to enhance model accuracy and performance metrics further using the UNSW NB-15 dataset.

REFERENCES

[1] M. De Donno, K. Tange, and N. Dragoni, “Foundations and Evolution of Modern Computing Paradigms: Cloud, IoT, Edge, and Fog,” IEEE Access, vol. 7, pp. 150936–150948, 2019, doi: 10.1109/ACCESS.2019.2947652.

[2] W. Zada Khan, E. Ahmed, S. Hakak, I. Yaqoob, and A. Ahmed, “Edge Computing: A Survey.”

[3] Y. Zhao, W. Wang, Y. Li, C. Colman Meixner, M. Tornatore, and J. Zhang, “Edge Computing and Networking: A Survey on Infrastructures and Applications,” IEEE Access, vol. 7, pp. 101213–101230, 2019, doi: 10.1109/ACCESS.2019.2927538.

[4] B. Varghese, N. Wang, S. Barbhuiya, P. Kilpatrick, and D. S. Nikolopoulos, “Challenges and Opportunities in Edge Computing,” in Proceedings - 2016 IEEE International Conference on Smart Cloud, SmartCloud 2016, Dec. 2016, pp. 20–26. doi: 10.1109/SmartCloud.2016.18.

[5] H. Zeyu, X. Geming, W. Zhaohang, and Y. Sen, “Survey on Edge Computing Security,” in Proceedings - 2020 International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering, ICBAIE 2020, Jun. 2020, pp. 96–105. doi: 10.1109/ICBAIE49996.2020.00027.

[6] S. Parikli, D. Dave, R. Patel, and N. Doshi, “Security and privacy issues in cloud, fog and edge computing,” in Procedia Computer Science, 2019, vol. 160, pp. 734–739. doi: 10.1016/j.procs.2019.11.018.

[7] A. Khraisat, I. Gondal, P. Vamplew, and J. Kamruzzaman, "Survey of intrusion detection systems: techniques, datasets, and challenges," Cybersecurity, vol. 2, no. 1, Dec. 2019, doi: 10.1186/s42400-019-0038-7.

[8] P. Spadaccino and F. Cuomo, “Intrusion detection systems for IOT: opportunities and challenges offered by edge computing and ML.”

[9] A. Thakkar and R. Lohiya, “Attack classification using feature selection techniques: a comparative study,” J Ambient Intell Humaniz Comput, vol. 12, no. 1, pp. 1249–1266, Jan. 2021, doi: 10.1007/s12652-020-02167-9.

[10] M. Ring, S. Wunderlich, D. Scheuring, D. Landes, and A. Hotho, “A Survey of Network-based Intrusion Detection Data Sets,” Mar. 2019, doi: 10.1016/j.cose.2019.06.005.

[11] S. Rajagopal, P. P. Kundapur, and K. S. Hareesha, “A Stacking Ensemble for Network Intrusion Detection Using Heterogeneous Datasets,” Security and Communication Networks, vol. 2020, 2020, doi: 10.1155/2020/4586875.

[12] A. Thakkar and R. Lohiya, “Attack classification using feature selection techniques: a comparative study,” J Ambient Intell Humaniz Comput, vol. 12, no. 1, pp. 1249–1266, Jan. 2021, doi: 10.1007/s12652-020-02167-9.

[13] S. M. Kasongo and Y. Sun, “Performance Analysis of Intrusion Detection Systems Using a Feature Selection Method on the UNSW-NB15 Dataset,” J Big Data, vol. 7, no. 1, Dec. 2020, doi: 10.1186/s40537-020-00379-6.

[14] S. S. Kareem, R. R. Mostafa, F. A. Hashim, and H. M. El-Bakry, “An Effective Feature Selection Model Using Hybrid Metaheuristic Algorithms for IoT Intrusion Detection,” Sensors, vol. 22, no. 4, Feb. 2022, doi: 10.3390/s22041396.

[15] N. Moustafa and J. Slay, “UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set),” in 2015 Military Communications and Information Systems Conference, MilCIS 2015 - Proceedings, Dec. 2015. doi: 10.1109/MilCIS.2015.7348942.

[16] UNSW-NB15 Dataset, UNSW Canberra Cyber 2015, https://www.unsw.adfa.edu.au/unswcanberra cyber/cybersecurity/ADFA-NB15-Datasets.

[17] A. Kumar, S. Kumar, V. Kumar, "Edge Computing based IDS Detecting Threats using Machine Learning and PyCaret, " *2023 International Conference on Computational Intelligence and Sustainable Engineering Solutions (CISES)*, IEEE, 2023.

[18] A. Kumar, V. Kumar, A. Saini, A. Kumari, V. Kumar, Classification of Minority Attacks using ML, International Conference on Fourth Industrial Revolution-based Technology and Practices (ICFIRTP), IEEE, 2022.
[19] Kumar, V., Kumar, V., Singh, N. et al. Enhancing Intrusion Detection System Performance to Detect Attacks on Edge of Things. SN COMPUT. SCI. 4, 802, 2023.