Advancing Network Intrusion Detection Systems with Machine Learning Techniques

Mourad Benmalek, Kamel-Dine Haouam

Advances in Artificial Intelligence and Machine Learning; Research 4 (3) 2575-2592 Received 11-07-2024; Accepted 19-09-2024; Published 26-09-2024

Advancing Network Intrusion Detection Systems with Machine Learning Techniques

Mourad Benmalek m_benmalek@yu.edu.sa
Computer Engineering Department, College of Engineering and Architecture
Al Yamamah University, Riyadh, Saudi Arabia

Kamel-Dine Haouam k_haouam@yu.edu.sa
Computer Engineering Department, College of Engineering and Architecture
Al Yamamah University, Riyadh, Saudi Arabia

Corresponding Author: Mourad Benmalek

Abstract

This paper presents an approach to enhancing the efficiency and effectiveness of Network Intrusion Detection Systems (NIDS) by leveraging Machine Learning (ML) techniques, specifically Decision Trees (DT), Naïve Bayes (NB), and Support Vector Machine (SVM). The proposed methodology involves a comprehensive evaluation and comparison of these algorithms using the NSL-KDD and UNSW-NB15 datasets, employing standard evaluation metrics such as accuracy, precision, recall, and F1-score. The study identifies the most effective algorithm for practical NIDS deployment. By providing actionable insights and recommendations for implementing the most suitable ML algorithm, this research contributes significantly to the ongoing efforts in strengthening network security against evolving cyber threats.

Keywords: Machine Learning (ML), Network Intrusion Detection Systems (NIDS), Decision Trees (DT), Naïve Bayes (NB), Support Vector Machine (SVM).

1. INTRODUCTION

Computer systems and data face significant threats from network intrusions, which involve unauthorized access, manipulation, or disruption of networks. Common threats include malware (viruses, worms, ransomwares, Trojans), phishing, DoS and DDoS attacks, Man-In-The-Middle (MITM) attacks, SQL injection, and zero-day exploits [1, 2]. To counter these, cybersecurity measures such as firewalls, Intrusion Prevention Systems (IPS), and Intrusion Detection Systems (IDS) are used.

IDS, a data mining-based system, examines network traffic to detect malicious activities through anomaly and signature-based techniques [3]. IDS systems also detect unauthorized access by monitoring suspicious activities, such as multiple failed logins that might indicate a brute force attack. By providing detailed alerts and information, IDS supports cybersecurity teams in effectively responding to breaches and improving incident responses [4]. IDS use two main approaches to address security concerns: signature-based and anomaly-based detection. Signature-based IDS relies on

Citation: Mourad Benmalek and Kamel-Dine Haouam. Advancing Network Intrusion Detection Systems with Machine Learning Techniques. Advances in Artificial Intelligence and Machine Learning. 2024;4(3):150.

predefined signatures of known threats, such as viruses and worms, and alerts administrators when observed network activity matches these signatures, providing high accuracy for known threats. In contrast, anomaly-based IDS establishes a baseline of normal behavior and flags deviations as potential risks, using statistical models to detect previously unknown threats [5].

Network Intrusion Detection Systems (NIDS) monitor packets in real-time, issuing alerts or automating responses to security risks, which enhances early detection and overall security [6]. NIDS, once reliant on rule-based and signature-based methods effective only against known threats, now benefit from the integration of Machine learning (ML). This enhancement allows NIDS to better adapt to dynamic network changes and classify threats more accurately. ML in NIDS includes supervised learning (using labeled data for known threats), unsupervised learning (detecting anomalies without pre-set labels), and reinforcement learning (adapting strategies based on feedback). The integration of ML and AI has significantly improved NIDS capabilities since 2017, with ongoing advancements further evolving these systems [6].

While anomaly-based systems can generate more false positives compared to signature-based ones, ML helps reduce these false positives by distinguishing genuine security concerns from regular variations and prioritizing incidents based on severity and context [7]. ML also enables these systems to adapt to network changes, enhancing their flexibility and accuracy in detecting malicious activities.

Evaluating the efficacy of NIDS is crucial as cyberattacks become more frequent and sophisticated. NIDS faces major challenges such as false negatives, which leave networks vulnerable, and false positives, which waste resources and impact system reliability. Signature-based detection often struggles with encrypted traffic and new attack patterns. ML offers a solution by analyzing large datasets to create adaptive models that respond dynamically to emerging threats, thus enhancing NIDS effectiveness and providing robust protection against advanced cyberattacks.

The primary objective of this paper is to investigate the application of ML techniques to enhance the efficiency and effectiveness of NIDS. The research aims to evaluate and compare the performance of three widely used ML algorithms: Decision Trees (DT), Naïve Bayes (NB), and Support Vector Machine (SVM) on two benchmark datasets (NSL-KDD and UNSW-NB15). By employing these algorithms, the study seeks to improve NIDS's ability to accurately detect and classify network intrusions while minimizing false positives and false negatives.

To assess the performance of the selected ML algorithms, the research will utilize standard evaluation metrics, including accuracy, precision, recall, and F1-score. These metrics will provide a comprehensive understanding of each algorithm's strengths and weaknesses in the context of NIDS. Furthermore, the study will explore the impact of various model parameters on the algorithms' performance, aiming to identify the optimal configurations for practical NIDS deployment.

The ultimate goal of this research is to provide actionable recommendations for implementing the most effective ML algorithm and model parameters in real-world NIDS applications. By doing so, this paper contributes to the ongoing efforts to strengthen network security and protect against the ever-evolving landscape of cyber threats.

The rest of this paper is organized as follows: Section 2 presents a review of related work. Section 3 provides an overview of NIDS datasets and discusses the background on the selected ML algorithms.

Section 4 describes the proposed methodology, including the datasets, algorithms, and evaluation metrics used in this study. Section 5 presents the analysis and discussion of the results, and Section 6 concludes the paper with a summary of the key findings and future directions.

2. RELATED WORK

Several pioneering studies have laid the groundwork for understanding the role of ML in NIDS. Bhati and Rai [8], evaluated various SVM techniques using the NSL-KDD dataset, highlighting Linear SVM, Quadratic SVM, Fine Gaussian SVM, and Medium Gaussian SVM with detection accuracies of 96.1%, 98.6%, 98.7%, and 98.5%, respectively. They discussed the necessity of security mechanisms due to the growth of information exchange and e-commerce, IDS components, and the challenges of training SVM with large datasets. Ma *et al.* [9], applied K-means, Farthest First, and COBWEB algorithms for clustering, and C-SVM for classification, achieving improved detection efficiency. Using data from the DARPA intrusion detection evaluation program, experiments included hierarchical clustering and SVM classification, measuring the classification accuracy of network attack descriptions.

According to [10], an accuracy rate of 99% was achieved on the NSL-KDD dataset when ten significant features were considered. NB achieved an accuracy rate of 83.6% after data pre-processing, with feature selection reducing training time and improving classification accuracy. Yassin *et al.* [11], used the ISCX 2012 dataset to assess the performance of an anomaly detection approach combining K-Means Clustering (KMC) with a NB Classifier (NBC). Results showed that the combination significantly improved accuracy, detection rates, and reduced false alarms. Ngueajio *et al.* [12], examined IDS using Support Vector Machines (SVM) on datasets like KDDCUP'99 and NSL-KDD, emphasizing Decision Trees' significant role, particularly when combined with methods like Simulated Annealing (SA). Decision Trees achieved up to 96% detection accuracy, and 98.6% when combined with other approaches, demonstrating their versatility in addressing complex cyber threats.

In another study, Karatas *et al.* [13], conducted a comparative analysis of various ML algorithms using the contemporary CSE-CIC-IDS2018 benchmark dataset. Their research tackled the prevalent issue of dataset imbalance by employing the Synthetic Minority Oversampling Technique (SMOTE). This approach effectively reduced the imbalance ratio, leading to a significant enhancement in the detection rates for less common attack classes. Their findings underscore the importance of addressing data imbalance in improving the overall effectiveness of intrusion detection systems. Yao *et al.* [14], proposed a Multilevel Semi-Supervised Machine Learning (MSML) framework for intrusion detection, combining clustering with RF models. Tested on the KDD Cup'99 dataset, their approach showed superior performance in detecting low-frequency attacks, demonstrating the potential of semi-supervised learning in identifying rare intrusion patterns. A comparative study by [15], evaluated multiple ML algorithms for intrusion detection using the KDDCUP'99 dataset. Employing 10-fold cross-validation to optimize performance, their results showed Decision Trees outperforming other classifiers with a 94 % accuracy rate.

Our study advances the field by: (1) comprehensively comparing SVM, NB, and DT algorithms across both NSL-KDD and UNSW-NB15 datasets, providing a more robust evaluation; (2) employing a wider range of performance metrics beyond accuracy alone; (3) offering actionable

recommendations for real-world NIDS deployment; and (4) bridging older and newer datasets to ensure relevance to current cybersecurity challenges.

3. BACKGROUND

This section provides an overview of prominent NIDS datasets and discusses the application of Support Vector Machine (SVM), Naïve Bayes (NB), and Decision Tree (DT) algorithms in NIDS.

3.1 Overview of NIDS Datasets

As shown in TABLE 1, three prominent datasets used in NIDS research are NSL-KDD, KDD99, and UNSW-NB15. NSL-KDD improves upon KDD99 by addressing its flaws, though KDD99 remains widely used due to its comprehensive range of attack types. UNSW-NB15 includes contemporary attack scenarios and offers a balanced distribution of instances across classes, facilitating robust IDS evaluation. The improvements in NSL-KDD over KDD99 validate its usefulness for developing effective anomaly detection techniques, while UNSW-NB15 provides a modern and nuanced dataset for network security research [16–19].

Table 1: Comparative Table of Network Intrusion Datasets.

	Purpose	Year	Records	Features	Classes
NSL-KDD	NIDS	2009	148,517	41	4 attack types
KDD99	NIDS	1999	4,898,431	41	14 attack types
UNSW-NB15	NIDS	2015	2,540,044	49	9 attack types

3.2 Support Vector Machine Algorithm

Support Vector Machines (SVMs) are widely used in machine learning for classification and regression across various fields [20]. They are crucial in evaluating and classifying network traffic for cybersecurity and are applied in assessing NIDS schemes, including various deep neural networks. However, there is a need for comprehensive surveys to scrutinize existing SVM-based NIDS approaches [21, 22]. One Against All (OAA) SVM involves training a dedicated SVM for each class, with classification achieved through majority voting [23]. SVMs aim to identify an optimal hyperplane that segregates data points into distinct classes [24]. The hyperplane serves as the decision boundary, intending to maximize the margin between classes [24]. Support Vector Regression (SVR) adapts SVMs for regression tasks, offering a nuanced approach to data variations [25].

3.3 Naïve Bayes Algorithm

Naïve Bayes (NB) is a well-known classification algorithm prized for its computational efficiency and wide application [26]. Various NB algorithms are used in different contexts, such as Gaussian

NB for continuous features, Multinomial NB for term frequency in documents, Bernoulli NB for binary features in text classification, and Complement NB for handling class imbalance [27, 28]. Naïve Bayes supports incremental learning, allowing the model to be updated with new data, which is useful in dynamic settings like changing network traffic characteristics [29]. However, it assumes feature independence, which is not always valid, and struggles with continuous features due to the assumption of a normal distribution [29]. Despite these limitations, it remains widely used and efficient, achieving an accuracy of 83.6% on the NSDL-KDD dataset [10].

3.4 Decision Tree Algorithm

Decision Trees (DT) excel in methodically analyzing large amounts of network data, distinguishing between normal and suspicious activities, and providing clear explanations for their decisions, aiding cybersecurity experts in both immediate responses and long-term system improvements. They consist of three node types: the root node, internal nodes, and leaf nodes. Gini impurity and information gain are key measures used in the process, quantifying the likelihood of incorrect classification and an attribute's ability to classify data by reducing entropy, respectively [30, 31]. Several notable algorithms enhance decision tree performance, including ID3, C4.5, CART, and Random Forests (RF) [32]. ID3 uses entropy and information gain for data splits but struggles with continuous variables and missing values. C4.5 improves upon ID3, handling both continuous and categorical data, and missing values, and includes post-pruning to prevent overfitting. CART supports both classification and regression tasks, using Gini impurity for classification and mean squared error for regression, with binary splits simplifying interpretation. RF, an ensemble method, builds multiple trees on bootstrapped data subsets, enhancing accuracy and reducing overfitting by averaging predictions. In NIDS, decision trees efficiently handle large datasets, excel in classification and prediction, adapt to evolving intrusion methods, and contribute to robust network security with their simplicity and speed [33].

4. PROPOSED METHODOLOGY

As shown in FIGURE 1, this section provides an overview of the data collection sources, preprocessing techniques, model development procedures, and evaluation metrics that will be utilized in constructing a robust ML-based NIDS.

We chose to evaluate three ML algorithms in this study: SVM, NB, and DTs. These algorithms were selected based on their popularity and effectiveness in previous studies on network intrusion detection. We aim to compare the performance of these algorithms on the NSL-KDD and UNSW-NB15 datasets and identify the most effective approach for NIDS.

4.1 Data Collection

This research will leverage two standard benchmark NIDS datasets: NSL-KDD and UNSW-NB15 based on their diversity, modern relevancy, and incorporation of contemporary attack scenarios unseen during model training.

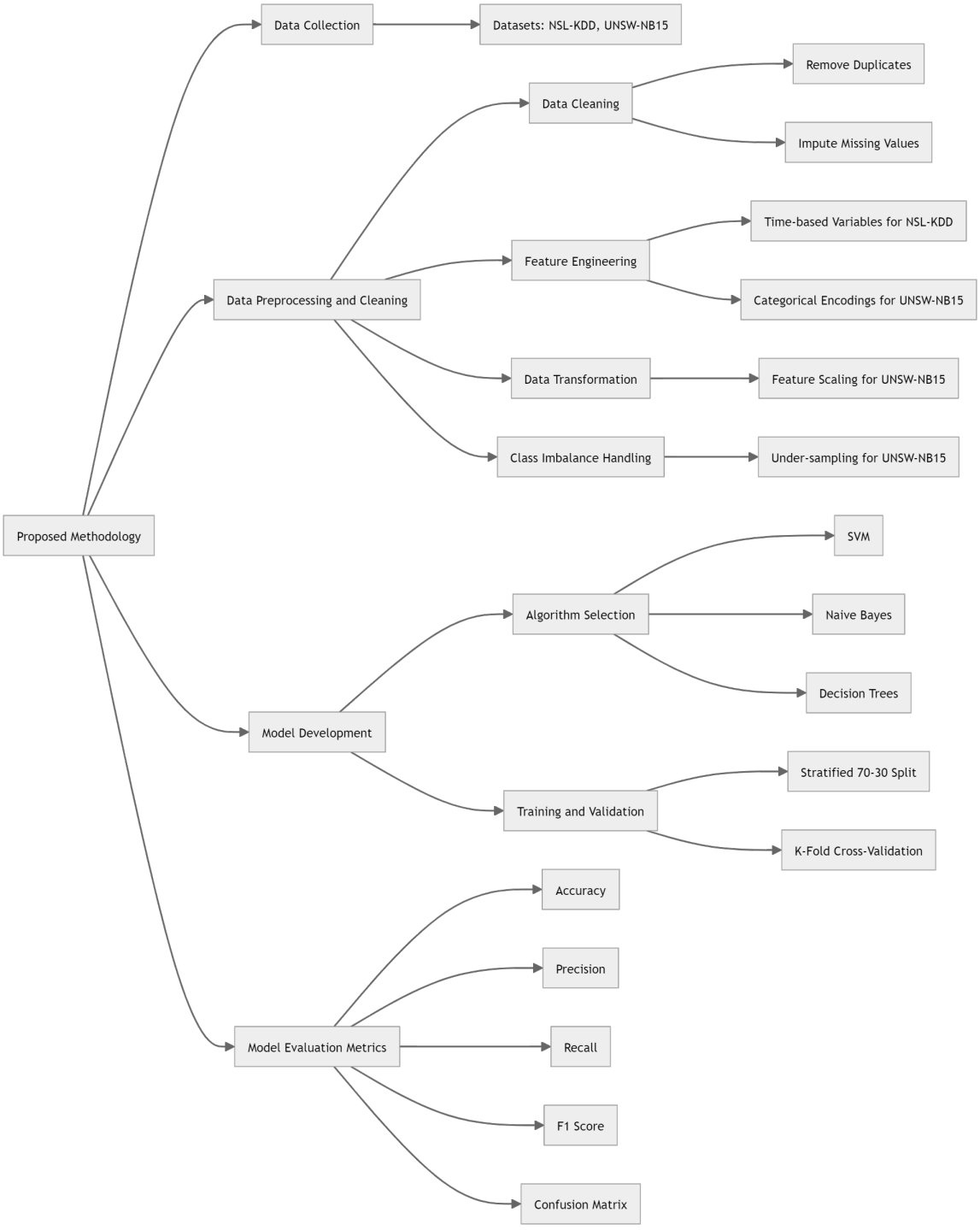

Figure 1: The Proposed Methodology

4.2 Data Preprocessing and Cleaning

The effectiveness of models is greatly influenced by data quality and preparation, beginning with data cleaning, which refines raw data into a suitable training dataset. Ensuring high-quality data involves addressing integrity issues like duplicate removal and missing value imputation. Creating new features can enhance a model's ability to learn patterns. For NSL-KDD, duplicates have been removed and no missing values exist. For UNSW-NB15, missing values are imputed with median values, and outliers are eliminated. Feature engineering includes generating new features such as time-based variables for NSL-KDD and categorical encodings for UNSW-NB15. Data transformation, including feature scaling for UNSW-NB15, prepares data for modeling. Class imbalance is addressed by under-sampling excess normal records in UNSW-NB15, while NSL-KDD is intrinsically balanced. Text data in UNSW-NB15 is label encoded, and rare categories are grouped. Both datasets are split into 70% training and 30% test sets using a stratified random split to maintain class proportions and ensure unbiased model evaluation.

4.3 Model Development

Model development is a critical phase that involves selecting an appropriate algorithm and training the model using a prepared dataset. Our dataset is divided into training, validation, and test sets to facilitate effective model training. Supervised machine learning is used due to labeled intrusion detection datasets, which allow models to learn decision boundaries between normal and malicious traffic. This study evaluates three popular supervised algorithms (SVM, NB, DT) based on their track records, ability to handle mixed data types, computational efficiency, and interpretability. These algorithms, along with their ensembles, are trained on a stratified 70% training and 30% test split, using k-fold cross-validation and grid search for hyperparameter tuning. The validation performance guides the selection of the best model for each algorithm.

4.4 Model Evaluation Metrics

The performance of a NIDS is assessed using several key metrics. These metrics collectively ensure a NIDS effectively distinguishes between normal and intrusive network activities, optimizing both detection accuracy and operational efficiency:

- **Accuracy**: The proportion of correct predictions out of all predictions.
- **Precision**: The ratio of true positives to the total number of positive predictions, indicating the system's reliability in identifying true threats.
- **Recall**: The ability to detect all actual intrusions, highlighting how effectively the system identifies real threats.
- **F1-Score**: The harmonic mean of precision and recall, providing a balanced measure, especially useful in imbalanced datasets.

- **Confusion Matrix**: Categorizes predictions into true positives, true negatives, false positives, and false negatives, offering a detailed breakdown of model performance and aiding system refinement.

5. ANALYSIS AND DISCUSSION

In this section, we assess the performance of three ML models (SVM, NB, and DT) in the context of network intrusion detection. We evaluate each model's effectiveness using metrics such as accuracy, recall, precision, and F1 score, supplemented by visual representations of confusion matrices to illustrate their classification capabilities.

5.1 SVM Model Results

As shown in TABLE 2, we evaluate the performance of the SVM model in classifying normal and abnormal network traffic for the NSL-KDD and UNSW-NB15 datasets, and its effectiveness in distinguishing various network attacks. For the NSL-KDD dataset, the SVM model demonstrated high accuracy (0.9896), recall (0.9895), precision (0.9894), and F1-Score (0.9896), with a confusion matrix showing strong detection capabilities and low misclassification rates. On the UNSW-NB15 dataset, the model's performance slightly decreased, with an accuracy of 0.9011, recall of 0.9011, precision of 0.9087, and F1-Score of 0.9013, indicating room for improvement in detecting intrusions.

Table 2: Performance Metrics of SVM Model on NSL-KDD and UNSW-NB15 Datasets.

	Accuracy	**Recall**	**Precision**	**F1-Score**
NSL-KDD	0.9896	0.9895	0.9894	0.9896
UNSW-NB15	0.9011	0.9011	0.9087	0. 9013

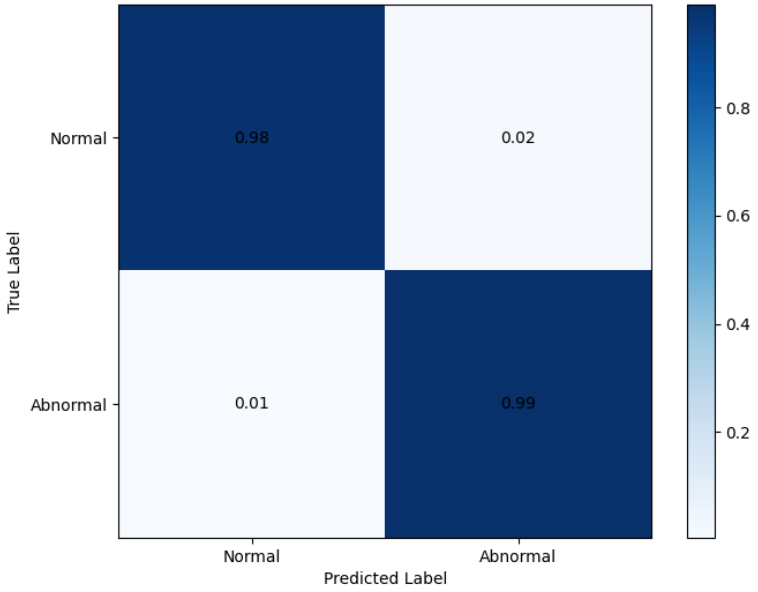

Figure 2: SVM Performance on NSL-KDD Dataset

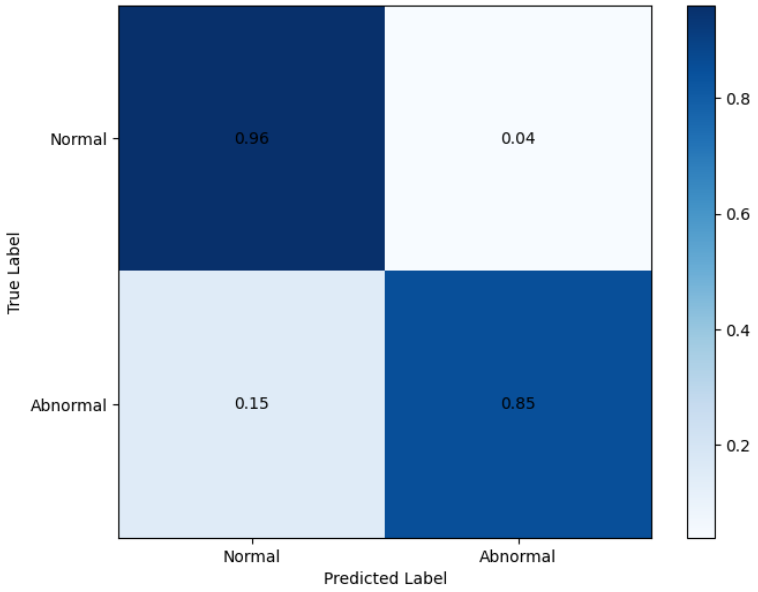

Figure 3: SVM Performance on UNSW-NB15 Dataset

The confusion matrix for the SVM model on the NSL-KDD dataset (FIGURE 2) reveals its strong performance, with 99% True Positives and 98% True Negatives. Low rates of False Positives (1.6%) and False Negatives (0.53%) indicate minimal unnecessary alerts and missed intrusions. On the UNSW-NB15 dataset (FIGURE 3), the model correctly identified 85% of abnormal instances (True Positives) and 96% of normal instances (True Negatives), but misclassified 4% of normal activities

(False Positives) and missed 15% of intrusions (False Negatives). This suggests the need for further optimization to improve its performance in network intrusion detection.

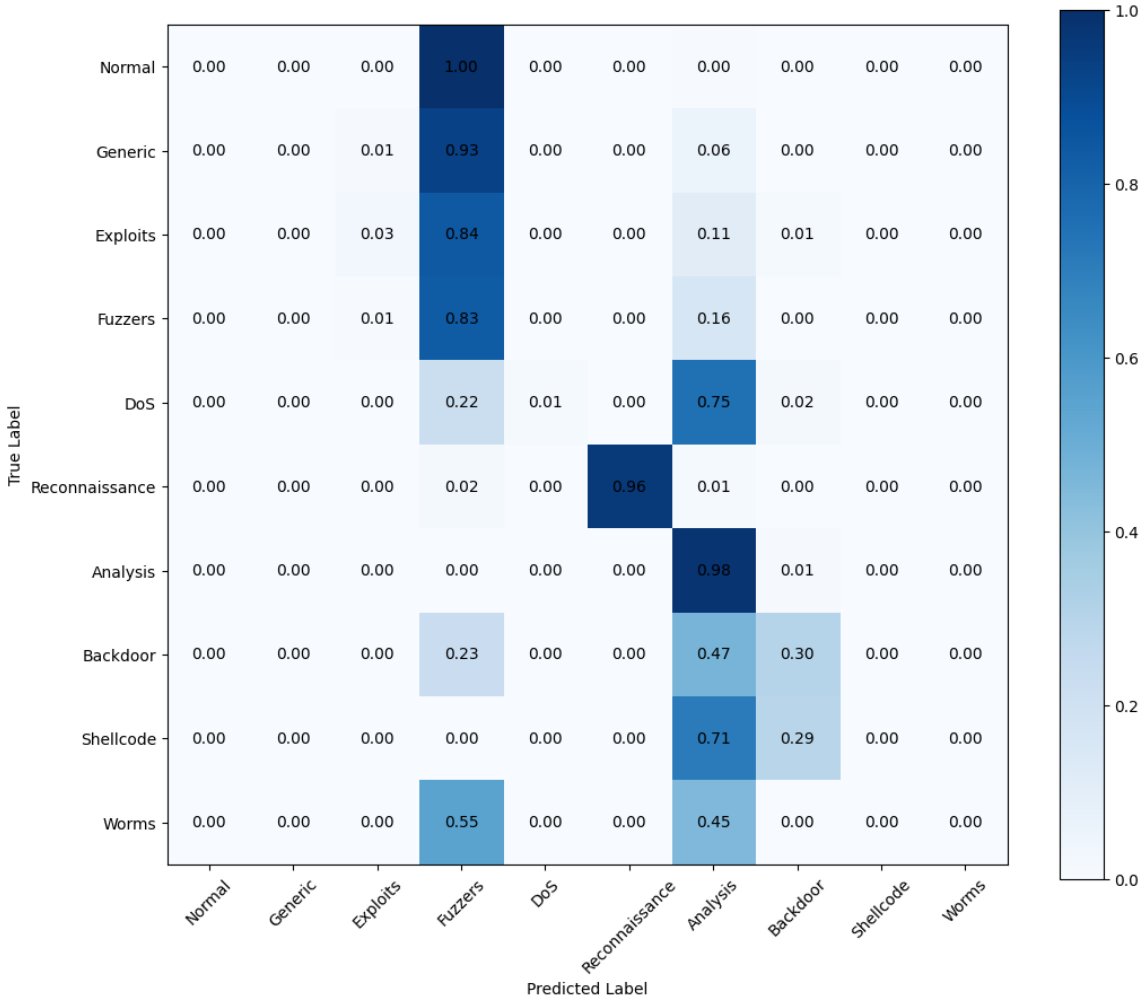

Figure 4: SVM Network Attack Classification Results

The confusion matrix for the SVM model on Network Attack (FIGURE 4) reveals its strong performance with high true positive rates for ”Reconnaissance” (96%), ”Generic” (93%), and ”Analysis” (98%) attacks. These high diagonal values indicate accurate classifications. However, the matrix also highlights misclassifications, such as ”DoS” attacks being wrongly classified as ”Normal” (0.014) and ”Reconnaissance” (0.022), and ”Backdoor” attacks as ”Analysis” (0.0028) and ”Shellcode” (0.3). While the SVM model shows high accuracy in identifying several intrusion types, these misclassifications suggest areas for improvement to enhance its overall effectiveness in network intrusion detection systems.

5.2 NB Model Results

As shown in TABLE 3, on the NSL-KDD dataset, the model achieved an accuracy of 0.9089, recall of 0.9089, precision of 0.9124, and an F1 score of 0.9084, indicating strong performance. On the UNSW-NB15 dataset, the model's performance was moderate, with an accuracy of 0.8003, recall of 0.8003, precision of 0.8112, and an F1 score of 0.7957, suggesting room for improvement.

Table 3: Performance Metrics of NB Model on NSL-KDD and UNSW-NB15 Datasets.

	Accuracy	Recall	Precision	F1-Score
NSL-KDD	0.9089	0.9089	0.9124	0.9084
UNSW-NB15	0.8003	0.8003	0.8112	0.7957

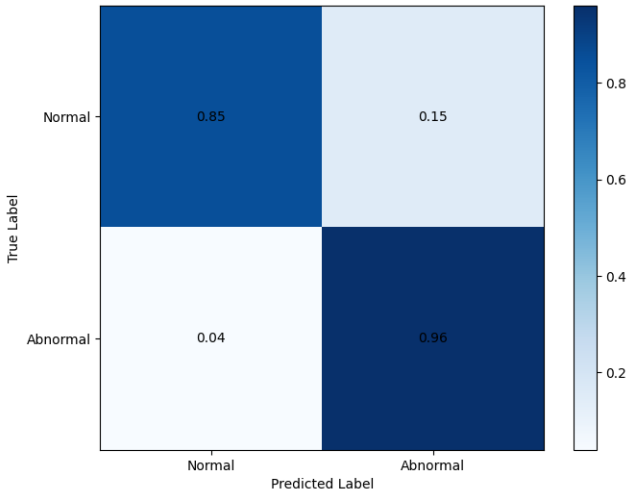

Figure 5: NB Performance on NSL-KDD Dataset

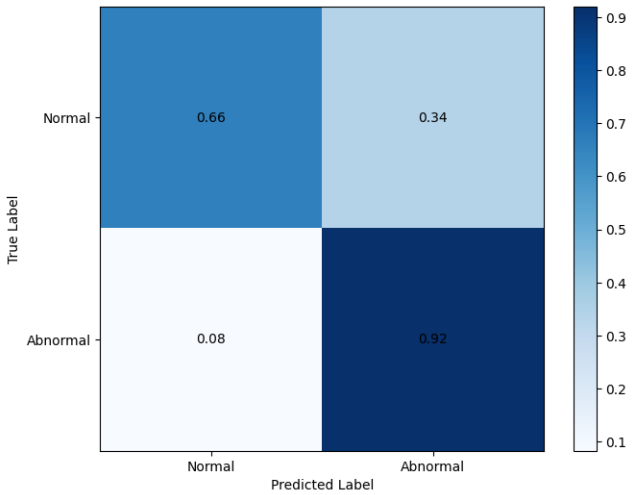

Figure 6: NB Performance on UNSW-NB15 Dataset

The confusion matrix for the NB model on the NSL-KDD dataset in FIGURE 5, shows it accurately identifies 96% of intrusions (True Positives) and correctly recognizes 85% of normal activities (True Negatives). However, it misclassifies 4% of normal activities as intrusions (False Positives) and misses 15% of actual intrusions (False Negatives). This indicates strong performance in True Positives and True Negatives but highlights areas for optimization to reduce False Positives and False Negatives. Similarly, the confusion matrix for the Naïve Bayes model on the UNSW-NB15 dataset in Figure 6, also shows a 96% identification rate for intrusions and an 85% recognition rate for normal activities, with the same rates of False Positives and False Negatives. This consistency underscores the need for improvements to enhance the model's accuracy and reliability in network intrusion detection.

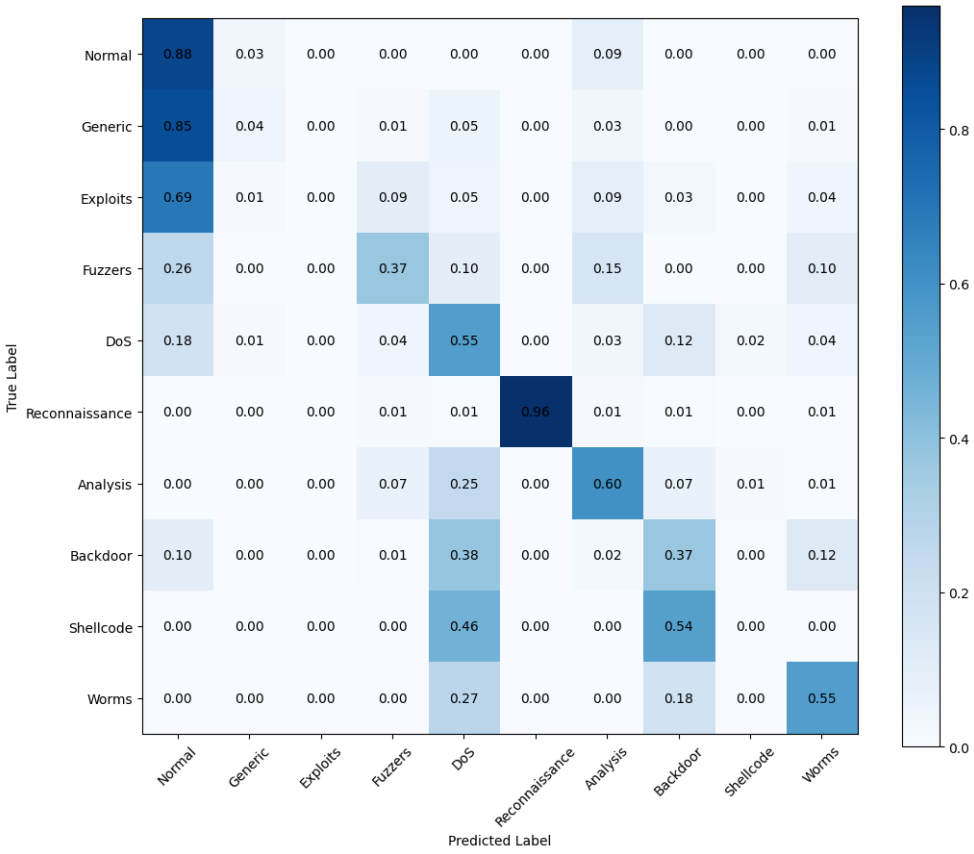

Figure 7: NB Network Attack Classification

The confusion matrix for the NB model (FIGURE 7) highlights its performance in classifying network attacks. The model achieved 88% accuracy for "normal" traffic, 85% for "generic" attacks, and 96% for "Reconnaissance" attacks, demonstrating strength in this area. However, misclassifications occurred, such as "Exploits" as "Normal" and "Fuzzers" (0.094) and "Backdoor" attacks misclassified as "Shellcode" (0.12) and "Reconnaissance" (0.094), indicating areas for improvement. Despite high accuracy for several intrusion types, these misclassifications suggest the need for further optimization.

5.3 DT Model Results

As shown on TABLE 4, on the NSL-KDD dataset, the model showed high reliability with an accuracy of 0.9867, recall of 0.9867, precision of 0.9868, and an F1 Score of 0.9867. For the UNSW-NB15 dataset, it achieved an accuracy of 0.9011, recall of 0.9011, precision of 0.9048, and an F1 Score of 0.9013, indicating good performance, though slightly lower than the NSL-KDD results.

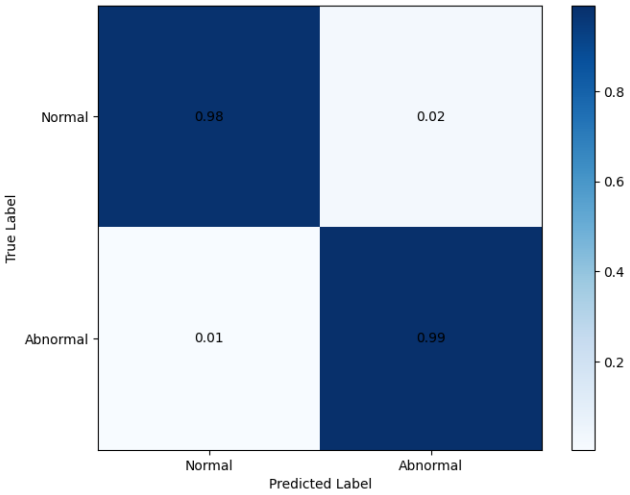

Figure 8: DT Performance on NSL-KDD Dataset

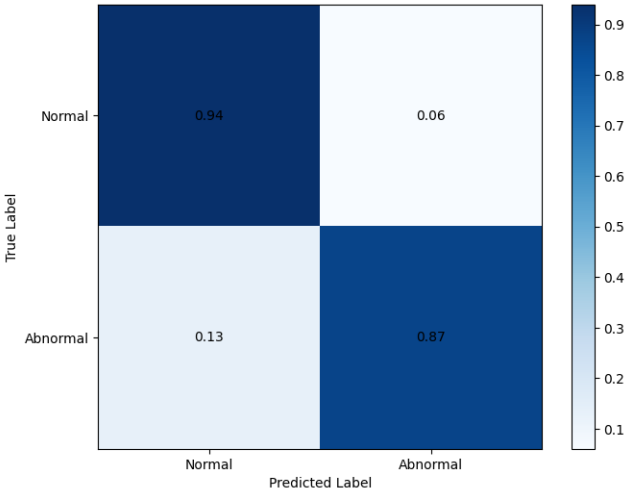

Figure 9: DT Performance on UNSW-NB15 Dataset

The confusion matrix for the DT model on the NSL-KDD dataset (FIGURE 8) demonstrates strong performance with 99% accuracy for intrusion detection (True Positives) and 98% for normal activities (True Negatives). Despite moderate false positives (2.3%) and minimal missed intrusions (0.51%), it shows high reliability. On the UNSW-NB15 dataset (FIGURE 9), the model maintains high accuracy in classifying normal activities (True Negatives at 94%) and intrusions (True Positives

at 87%), but struggles with a higher rate of false positives (6.2%) and significant false negatives (13%), suggesting a need for further optimization in intrusion detection.

Table 4: Performance Metrics of DT Model on NSL-KDD and UNSW-NB15 Datasets.

	Accuracy	**Recall**	**Precision**	**F1-Score**
NSL-KDD	0.9867	0.9867	0.9868	0.9867
UNSW-NB15	0.9011	0.9011	0.9048	0.9013

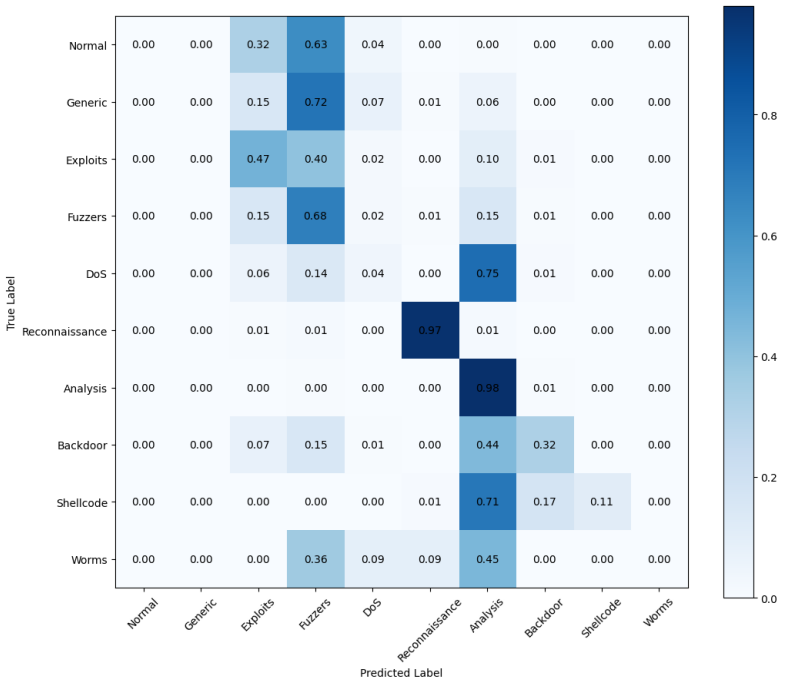

Figure 10: DT Network Attack Classification

As shown in Figure FIGURE 10, The DT model's confusion matrix demonstrates strong performance in identifying "reconnaissance" (97%), "DoS" (75%), and "analysis" (98%) attacks, but also reveals misclassifications such as "Exploits" being mislabeled as "Normal" and "Fuzzers," and "backdoor" attacks confused with "analysis" and "shellcode." These errors highlight areas for improvement, suggesting that while the model is effective overall, further refinement is needed to enhance its accuracy in detecting network intrusions.

5.4 Comparative Analysis for Algorithms Performance

We compared the performance of SVM, NB, and DT algorithms for network intrusion detection using the NSL-KDD and UNSW-NB15 datasets. Evaluation metrics including accuracy, recall, precision, and F1 score were calculated and visualized using histograms in FIGURE 11 and FIGURE 12.

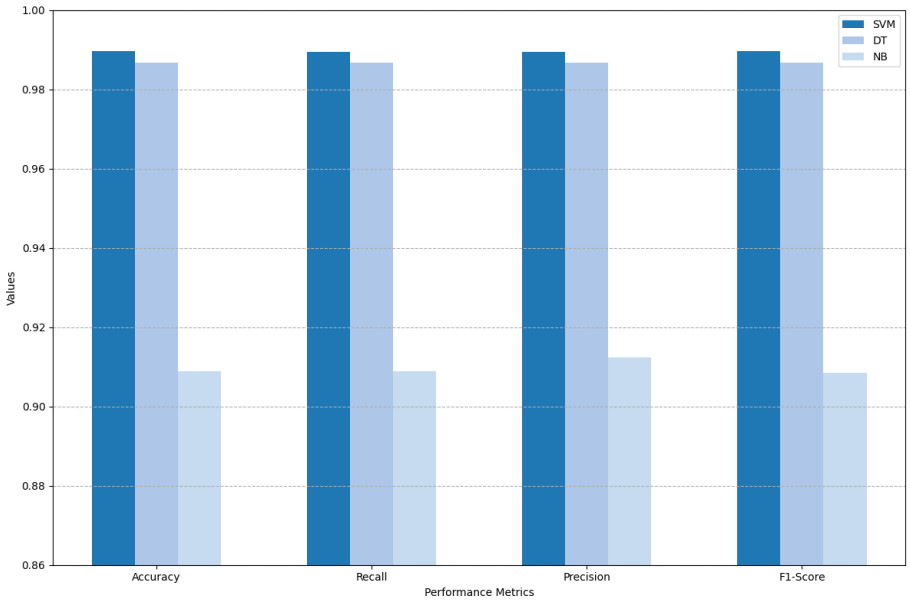

Figure 11: Performance comparison of SVM, NB, and DT algorithms on the NSL-KDD dataset

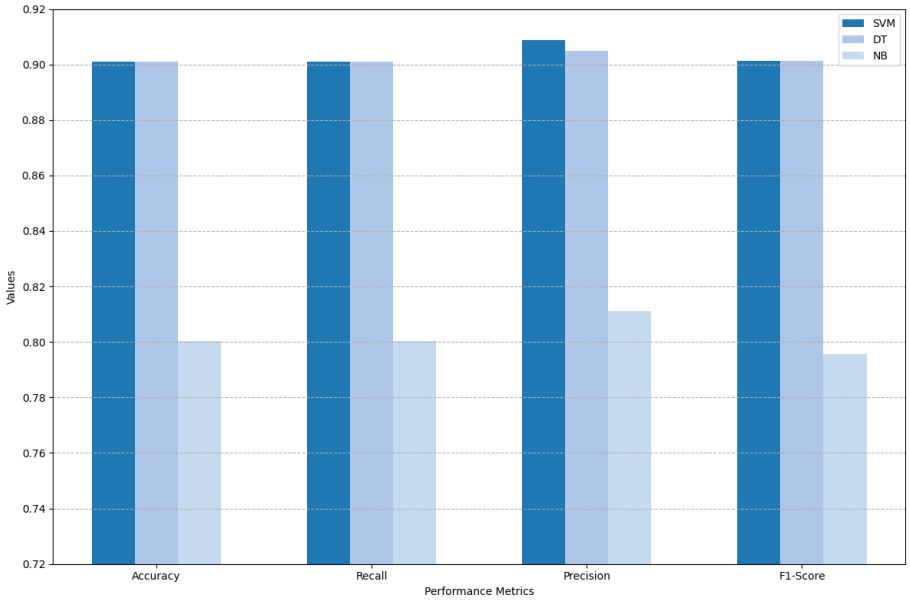

Figure 12: Performance comparison of SVM, NB, and DT algorithms on the UNSW-NB15 dataset

On the NSL-KDD dataset, the SVM and DT algorithms demonstrated superior performance compared to NB, achieving metric scores of approximately 0.98. In contrast, the NB algorithm had lower scores below 0.90 for the measured metrics. The strong results from SVM and DT suggest that the separation of normal and anomalous patterns in the NSL-KDD feature space is more suited for the maximum-margin hyperplane approach of SVM and the rule-based, tree-like decision structure of DT. The NB algorithm, which is based on probabilistic assumptions of feature independence,

appears less able to model the underlying data distribution effectively, leading to more misclassifications.

Similarly, on the UNSW-NB15 dataset, the SVM and DT algorithms performed well, with metric scores close to 0.90. However, the NB algorithm again showed weaker results, with scores below 0.80 for the evaluated metrics. These findings indicate that the NB algorithm is less effective at capturing the complex relationships and dependencies in the UNSW-NB15 dataset. The dataset's features may exhibit correlations or non-linear interactions that violate the Naïve independence assumption of the NB classifier. On the other hand, the SVM's kernel trick and the DT's ability to learn hierarchical decision boundaries seem more robust to these dataset characteristics.

The main reasons for the performance differences can be attributed to the inherent assumptions and strengths of each algorithm. SVM is known for its ability to handle high-dimensional data and find optimal decision boundaries, making it effective for intrusion detection tasks where distinguishing between normal and anomalous patterns is crucial. DTs can learn interpretable rule-based models that can capture complex relationships in the data. In contrast, the NB algorithm's assumption of feature independence may be too simplistic for the intrusion detection domain, where features often exhibit correlations and dependencies.

6. CONCLUSION

Our research evaluated SVM, NB, and DT models using NSL-KDD and UNSW-NB15 datasets to enhance NIDS. SVM consistently outperformed the others across all metrics, followed by DT, while NB showed lower effectiveness. These findings highlight the potential of ML-based approaches in improving NIDS capabilities, with SVM emerging as a promising candidate for real-world implementation. However, there is room for improvement in reducing false positives and negatives. The study's results suggest that integrating ML models, particularly SVM, into existing NIDS frameworks could substantially boost threat detection capabilities. However, the varying performance across datasets emphasizes the need for continuous model evaluation and adaptation to evolving threats. Future work will focus on optimizing models through advanced techniques, exploring ensemble methods and other ML algorithms, combining diverse datasets, and testing in real-world environments. These efforts aim to further enhance the accuracy, efficiency, and adaptability of ML-based NIDS, contributing to more resilient network security in an increasingly complex digital landscape.

References

[1] Latha S, Prakash SJ. A Survey on Network Attacks and Intrusion Detection Systems. In: 4th international conference on advanced computing and communication systems (ICACCS). IEEE PUBLICATIONS. 2017;2017:1-7.

[2] Benmalek M. Ransomware on Cyber-Physical Systems: Taxonomies, Case Studies, Security Gaps, and Open Challenges. Internet Things Cyber-Phys Syst. 2024;4:186-202.

[3] He K, Kim DD, Asghar MR. Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey. IEEE Commun Surv Tutorials. 2023;25:538-566.

[4] Heidari A, Jabraeil Jamali MA. Internet of Things Intrusion Detection Systems: A Comprehensive Review and Future Directions. Clust Comput. 2023;26:3753-3780.

[5] Zipperle M, Gottwalt F, Chang E, Dillon T. Provenance-Based Intrusion Detection Systems: A Survey. ACM Comput Surv. 2022;55:1-36.

[6] Vanin P, Newe T, Dhirani LL, O'Connell E, O'Shea D, et al. A Study of Network Intrusion Detection Systems Using Artificial Intelligence. Applied Sciences. 2022;12:11752.

[7] Apruzzese G, Pajola L, Conti M. The Cross-Evaluation of Machine Learning-Based Network Intrusion Detection Systems. IEEE Trans Netw Serv Manage. 2022;19:5152-5169.

[8] Bhati BS, Rai CS. Analysis of Support Vector Machine-Based Intrusion Detection Techniques. Arab J Sci Eng. 2019;45:2371-2383.

[9] Ma T, Wang F, Cheng J, Yu Y, Chen X. A Hybrid Spectral Clustering and Deep Neural Network Ensemble Algorithm for Intrusion Detection in Sensor Networks. Sensors. 2016;16:1701.

[10] Abrar I, Ayub Z, Masoodi F, Bamhdi AM. A Machine Learning Approach for Intrusion Detection System on Nsl-Kdd Dataset. In: 2020 International Conference on Smart Electronics and Communication (ICOSEC). IEEE PUBLICATIONS. 2020:919-924.

[11] https://soc.uum.edu.my/icoci/2023/icoci2013/PDF/PID49.pdf

[12] Ngueajio MK, Washington G, Rawat DB, Ngueabou Y. Intrusion Detection Systems Using Support Vector Machines on the Kddcup'99 and Nsl-Kdd Datasets: A Comprehensive Survey. In: Proceedings of the Sai Intelligent Systems Conference. Springer International Publishing. 2022:609-629.

[13] Karatas G, Demir O, Sahingoz OK. Increasing the Performance of Machine Learning-Based Idss on an Imbalanced and Up-To-Date Dataset. IEEE Access. 2020;8:32150-32162.

[14] Yao H, Fu D, Zhang P, Li M, Liu Y. MSML: A Novel Multilevel Semi- Supervised Machine Learning Framework for Intrusion Detection System. IEEE Internet Things J. 2018;6:1949-1959.

[15] Alqahtani H, Sarker IH, Kalim A, Minhaz Hossain SM, Ikhlaq S, et al. Cyber Intrusion Detection Using Machine Learning Classification Techniques. In: Chaubey, N., Parikh, S., Amin, K. (eds) Computing Science, Communication and Security. COMS2 2020. Communications in Computer and Information Science. Springer Singapore. 2020;1235.

[16] Al-Daweri MS, Zainol Ariffin KA, Abdullah S, Md. Senan MFE. An Analysis of the KDD99 and Unsw-NB15 Datasets for the Intrusion Detection System. Symmetry. 2020;12:1666.

[17] Janarthanan T, Zargari S. Feature Selection in UNSW-NB15 and KDDCUP'99 Datasets. In 2017 IEEE 26th International Symposium on Industrial Electronics (ISIE) IEEE. 2017:1881-1886.

[18] Moustafa N, Slay J. The Significant Features of the UNSW-NB15 and the KDD99 Data Sets for Network Intrusion Detection Systems. In 2015 4th International Workshop on Building Analysis Datasets and Gathering Experience Returns for Security. IEEE. 2015:25-31.

[19] Jing D, Chen HB. Svm Based Network Intrusion Detection for the Unsw-NB15 Dataset. In 2019 IEEE 13th International Conference On ASIC (ASICON). IEEE.2019:1-4.

[20] Hearst MA, Dumais ST, Osuna E, Platt J, Scholkopf B. Support Vector Machines. IEEE Intelligent Systems and Their Applications. 1998:13:18-28.

[21] Khraisat A, Gondal Vamplew P, Kamruzzaman J. Survey of Intrusion Detection Systems: Techniques, Datasets and Challenges. Cybersecurity. 2019;2:1-22.

[22] Mohammadi M, Rashid TA, Karim SH, Aldalwie AH, Tho QT, et al. A Comprehensive Survey and Taxonomy of the Svm-Based Intrusion Detection Systems. J Network Comput Appl. 2021;178:102983.

[23] Duan Y, Zou B, Xu J, Chen F, Wei J, Tang YY. OAA-SVM-MS: A Fast and Efficient Multi-Class Classification Algorithm. Neurocomputing. 2021;454:448-460.

[24] Cervantes J, Garcia-Lamont F, Rodríguez-Mazahua L, Lopez A. A Comprehensive Survey on Support Vector Machine Classification: Applications, Challenges and Trends. Neurocomputing. 2020;408:189-215.

[25] Zhang F, O'Donnell LJ. Support Vector Regression. In: Machine Learning. Academic Press. 2020:123-140.

[26] Rish I. An Empirical Study of the Naive Bayes Classifier. In: IJCAI 2001 Workshop on Empirical Methods in Artificial Intelligence. 2001;3:41-46.

[27] Xu S. Bayesian Naïve Bayes Classifiers to Text Classification. J Inf Sci. 2018;44:48-59.

[28] Seref B, Bostanci E. Performance Comparison of Naïve Bayes and Complement Naïve Bayes Algorithms. In: 6th International Conference on Electrical and Electronics Engineering (ICEEE). 2019:131-138.

[29] Sharma SK, Pandey P, Tiwari SK, Sisodia MS. An Improved Network Intrusion Detection Technique Based on K-Means Clustering via Naïve Bayes Classification. In IEEE-International Conference on Advances in Engineering, Science and Management (ICAESM-2012). IEEE Publications. 2012:417-422.

[30] De Ville B. Decision Trees. Wiley Interdiscip Rev Comput Stat. 2013;5:448-455.

[31] Haouam KD, Benmalek M. Machine Learning Algorithms for Early Prediction of Multiple Sclerosis Progression: A Comparative Study. Adv Artif Intell Mach Learn. 2024;4:2027-2051.

[32] Bashir S, Qamar U, Khan FH, Javed MY. An Efficient Rule-Based Classification of Diabetes Using ID3, C4. 5,CART ensembles. In: 12th International Conference on Frontiers of Information Technology,CART Ensembles. IEEE Publications. 2014:226-231.

[33] Magán-Carrión R, Urda D, Díaz-Cano I, Dorronsoro B. Towards a Reliable Comparison and Evaluation of Network Intrusion Detection Systems Based on Machine Learning Approaches. Appl Sci. 2020;10:1775.