Available online www.unicrossjournals.com

UNICROSS JOURNAL OF SCIENCE AND TECHNOLOGY, UJOST

RESEARCH ARTICLE VOL. 3(3) SEPTEMBER, 2024 ISSN:2814-2233

Date Approved September 30, 2024 Pages 214 - 222

DEVELOPMENT AND APPLICATION OF ADVANCED SUPPORT VECTOR MACHINE ALGORITHM WITH RADIAL BASIS FUNCTIONS FOR DETECTING DDoS ATTACKS.

Anthony Otiko,
otikotony@gmail.com; otikotony@unicross.edu.ng
Department of Computer Science, University of Cross River State, Calabar.

Abstract

This study explores the application of Support Vector Machine (SVM) algorithms for Distributed Denial-of-Service (DDoS) attack detection using the KDD-99 dataset from Kaggle's machine learning repository. The research investigates the performance of SVM models across various kernels as part of hyperparameter tuning, including sigmoid, polynomial, linear, and radial basis function kernels. The methodology involves three key phases: data preprocessing, classification using SVM, and performance evaluation. Data preprocessing includes tasks such as normalization and feature standardization. SVM models are trained on preprocessed data, and their performance is evaluated using metrics such as F1-score, precision, recall, and accuracy. Results show significant performance variations among SVM kernels, with the radial basis function kernel demonstrating robust performance with 99% accuracy and the sigmoid kernel exhibiting subpar results. This study emphasizes the importance of selecting an appropriate SVM kernel based on dataset characteristics for effective DDoS attack detection and network security enhancement. Further research could explore advanced techniques to improve model performance against evolving cyber threats.

Keywords: Attacks, Algorithms, Detection, DDoS, Networks, Machine Learning.

1.0 Introduction

In the contemporary digital sphere, network services face a persistent threat from various sources, ranging from intentional assaults such as Denial-of-Service (DOS) attacks to unintended consequences like Flash Events (FE). DOS attacks, notably Distributed Denial-of-Service (DDOS) attacks, disrupt normal internet traffic patterns, posing significant challenges to online operations (Osanaiye *et al.*, 2016). The prevalence of the

DOS issue traces back to the 1970s, with notable incidents such as the Morris Worm in

1998 and subsequent large-scale attacks in 1999 targeting commercial websites.

Essentially, a Distributed Denial of Service (DDoS) attack represents a potent method for disrupting both intranet and internet resources. It operates through a straightforward yet highly impactful technique, wherein legitimate users are effectively prevented from accessing web-based services due to the concerted efforts of numerous compromised machines (Singh and Gupta, 2022). DDoS attacks manifest across network, transport, and application layers, employing various protocols such as TCP, UDP, ICMP, and HTTP to execute their disruptive actions (Alharbi et al., 2021). These events marked pivotal moments in the evolution of the threat landscape, highlighting the need for robust defense mechanisms against malicious activities in cyberspace.

However, numerous researchers have endeavored to address network security challenges through various methodologies, including Machine Learning (ML), knowledge-based approaches, and statistical analysis. However, each method presents its unique set of limitations and challenges. For instance, statistical methods often struggle to precisely determine the normal distribution of network packets (Sharafaldin *et al.*, 2019). While ML techniques offer promising results, however, the optimal selection of the best algorithm for the problem in concern remains a primary concern.

SVM classifiers have gained attention in cybersecurity due to their ability to effectively classify complex and high-dimensional data, making them suitable for detecting anomalous network traffic associated with DDoS attacks (Gaye *et al.*, 2021). However, the selection of the best parameter via the kernel can be challenging.

Therefore, this study proposes the utilization of the Support Vector Machine (SVM) on the KDD dataset obtained from the Kaggle machine learning repository. The investigation delves into evaluating the model's performance across various kernels as part of hyperparameter tuning. These kernels encompass the sigmoid, polynomial, linear, and radial basis function kernels.

The primary contribution of this study lies in the evaluation of the performance of various SVM kernels for classifying network attacks and proposing an approach to mitigate DDoS attacks. The remainder of this paper is structured as follows: Section 2 presents the literature review, encompassing various related surveys and the methodology employed for the literature review. In Section 3, the methodology for applying the SVM algorithm is discussed in detail. Section 5 presents the results of applying the SVM algorithm using different kernels. Finally, the conclusion of the study is outlined in the concluding chapter.

2.0 Related works

In their study, Malik and Aslam (2013) put out a hybrid approach that combines multi-objective particle swarm optimization with the random forest algorithm. They suggest employing this method to effectively identify Probe attacks. Their goal is to enhance the detection rate and reduce the rate of false alarm discovery while identifying Probe attacks. Particle Swarm Optimization (PSO)

removes superfluous features, whereas Random Forest (RF) identifies Probe assaults. The proposed approach has a detection rate of 90.7%.

Li et al., (2018) constructed a model that incorporates the Gini index. This model comprises the gradient boosting decision tree (GBDT) and particle swarm optimization (PSO). The Gini index was used to select the best feature subset. The network assault was detected using the gradient lifting decision tree algorithm. The GBDT parameters were optimized using the Particle Swarm Optimization (PSO) technique. The model was evaluated based on its detection rate, accuracy, F1-score, precision, and false alarm rate. An evaluation was carried out using the NSL-KDD Dataset. According to the findings, the model demonstrated high accuracy and effective intrusion detection capabilities. The model achieved a detection rate of 78.48%, a precision rate of 96.44%, an F1-score of 86.54%, and a false acceptance rate of 3.83%.

Yin et al., (2017) sought to enhance the efficiency of a Network Intrusion Detection System (NIDS) by optimizing the performance of a Recurrent Neural Network (RNN) based approach. Additionally, the researchers conducted model training on the NSL-KDD dataset, evaluating both binary and multi-class classification tasks. According to Yin et al. (2017), the performance of the Recurrent Neural Network (RNN) in the context of the Intrusion Detection System (IDS) exhibited superiority in both classification tasks when compared to conventional techniques. The authors assert that the utilization of Recurrent Neural Networks (RNN) in Intrusion Detection Systems (IDS) exhibits a robust modelling capability, in contrast to the recommended Software-Defined Networking (SDN) environment put forth by (Tang *et al.*, 2016). Therefore, the model developed by the author was only trained using the six fundamental characteristics extracted from the NSL-KDD dataset, employing various learning rates. As a result, the model attains a peak accuracy of 75.75%.

In their study, Yan et al. (2018) empirically analyzed the performance of four deep learning models multi-layer perceptron (MLP), restricted Boltzmann machine (RBM), sparse autoencoder (SAE), as well as the combined model of wide linear classifier and deep neural network (WND). The analysis is conducted on two intrusion datasets, namely NSL-KDD and UNSW-NB15. Nevertheless, their tests fail to encompass certain widely used models such as recurrent neural networks, and they neglect to do any assessment of more recent intrusion datasets. In addition, the authors solely present the accuracy, precision, and recall of the trials. The SAE algorithm achieves the maximum accuracy of 79.3% on the NSL-KDD dataset, while the WND algorithm achieves the highest accuracy of 91.2% on the UNSW-NB15 dataset.

3.0 Research methodology

The proposed research methodology encompasses three principal phases: data preprocessing, classification, and performance evaluation. In detail, the data preprocessing phase aims to refine and cleanse the raw data for subsequent analysis.

This stage involves tasks such as handling missing values, normalizing data, and eliminating noise or outliers. The primary objective is to ensure the data is suitably formatted for further analysis. The subsequent phase entails classification, where a Support Vector Machine (SVM) is employed to construct a model using the preprocessed training dataset using the different kinds of SVM kernels. The final stage denoted as the performance evaluation phase, involves assessing the efficacy of the SVM algorithm utilizing an independent test dataset. This evaluation encompasses various performance metrics, including recall, precision, and accuracy score, which collectively gauge the model's reliability and effectiveness in identifying network intrusions. Through this phase, the robustness and performance of the SVM model are rigorously scrutinized to ascertain its suitability for practical deployment in intrusion detection systems. The visualization of the proposed methodology approach is depicted in Figure 3.2 below, offering a comprehensive overview of the sequential workflow adopted throughout the research endeavor.

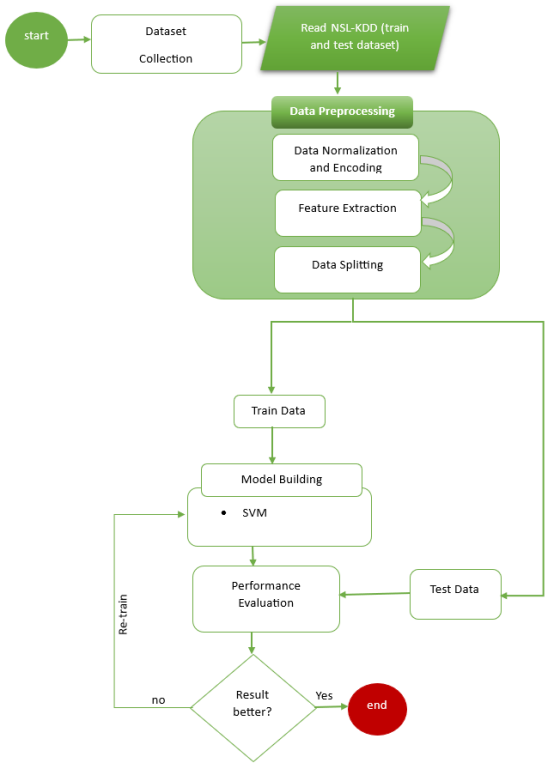

Figure 3.1: Research Design

3.1 Dataset description

The KDD dataset has 41 attributes and one class label with more than 4 million instances (Ibor et al., 2015; Aggarwal and Sharma, 2015). The dataset has several versions from which 10% of the training data with 494, 021 instances is used in this empirical study. The test set contains 311, 079 instances, which are used to evaluate the performance of the model for accuracy, FPR and other metrics. The dataset also has a class label that indicates the type of attack for each instance. Different configurations of this dataset are available with variation in the number of instances but the number of attributes in each case is 42. The attribute labeled 42 in the data set is the class attribute which indicates whether a given instance is a normal connection instance or an attack.

3.2 Data preprocessing

The data preprocessing phase is essential to address the disparate range of values exhibited by each characteristic in the intrusion detection data. This phase involves characterizing the data, which includes converting text features to normal values and standardizing numerical features. Specifically, the numerical characteristics of the intrusion data are standardized using the formula outlined in this paper.

$$\frac{V^{ij} - min^{j}(V^{ij})}{max^{j}(V^{ij}) - min^{j}(V^{ij})} \dots\dots\dots\dots\dots\dots 3.1$$

This objective is accomplished by normalizing each feature variable, denoted as Vij, to the interval (0, 1).

3.2 Classification algorithm

Considering that the problem of DDOS attack from the sourced dataset is identified to be a classification problem. This study addresses the challenge of identifying Distributed Denial of Service (DDOS) attacks using a classification approach. To tackle this issue, the research proposes employing the Support Vector Machine (SVM) algorithm. SVM is a well-established machine-learning technique primarily used for classification tasks. Rooted in the concept of solving large-dimensional problems' dual forms, SVM aims to construct a classifier based on a minimal set of support vectors, thereby adhering to the principle of structural risk minimization (Gaye et al., 2021). Functioning as a classical two-classification model, SVM operates by determining an optimal hyperplane to partition the dataset into distinct categories. This partitioning principle involves maximizing the margin, encompassing both hard and soft margins, and is typically addressed through quadratic programming optimization techniques.

3.3 Performance evaluation

To evaluate the performance of the SVM model, the accuracy, precision, recall, and f1-score metrics are proposed. The criterion for each of the metrics is calculated according to four main criteria which are True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) as follows:

i. True Positive (TP): Indicates when an alarm is generated and there is an intrusion.
ii. False Negative (FN): Indicates when an alarm is not generated but there is an intrusion.
iii. False Positive (FP): Indicates when an alarm is generated but there is no intrusion.
iv. True Negative (TN): Indicates when an alarm is not generated and there is no intrusion.

Accuracy is a frequently employed metric for assessing the comprehensive performance of a classification model. It quantifies the ratio of accurately classified instances, encompassing both true positives and true negatives, relative to the total number of instances, as demonstrated in equation 3.2:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \dots\dots\dots\dots\dots\dots 3.2$$

Precision, as a metric, centres on the positive class and evaluates the accuracy of predicted positives. It addresses the inquiry: "Out of all instances predicted as positive, how many were truly positive?". Equation 3.3 illustrates the mathematical representation of the precision metric.

$$precision = \frac{TP}{TP + FP} \dots\dots\dots\dots\dots\dots\dots\dots 3.3$$

Recall: Recall, also known as sensitivity or true positive rate, evaluates how many of the actual intrusion records were correctly predicted. It answers the question: "Of all the actual positives, how many were correctly predicted as positive?" The formula is shown in equation 3.4:

$$Recall = \frac{TP}{TP + FN} \dots\dots.. 3.4$$

F-measure, also known as F-score, represents the harmonic mean of precision and recall. It offers a well-rounded assessment of a classification model, considering both false positives and false negatives. This metric is especially valuable when analyzing imbalanced datasets. Equation 3.5 presents the mathematical expression for calculating the F-score.

$$F - Measure = 2 \times \frac{Precision \times Recall}{Precision + Recall} \dots\dots\dots..3.5$$

4.0 Result and discussion

This section outlines the findings from an empirical study and application that aimed to forecast network anomaly detection using the KDD-99 dataset source from the Kaggle machine learning repositories. The use of machine learning algorithms, specifically the SVM algorithm is part of the study technique. The developed models are then thoroughly validated using performance metrics including F1-score, precision, and recall.

4.1 Environmental Setup

The Anaconda programming environment's computational resilience was utilized in the creation of the network DDOS model. The complexities of the model were executed on a Windows operating system with a dual-core Intel Core i5 processor and 4GB RAM, and they were smoothly integrated into the Python programming language. Implementation was done using SKlearn API, which is designed for complex deep neural network applications. Important Python modules like NumPy, in addition to SKlearn, enabled sophisticated numerical operations, and pandas took care of dataset integration and parsing. A flexible visualization application called Matplotlib was used to show the graphical behaviour of painstakingly constructed machine learning models.

4.3 Result presentation

The results of the SVM model on the KDD dataset for different SVM kernels reveal significant variations in performance metrics across different kernel types as presented in Table 4.1. For the Linear kernel, the model demonstrates strong performance in terms of F1-Score, Precision, and Recall, achieving values of 0.87, 0.78, and 0.99 respectively. However, the Accuracy is slightly lower at 0.80 compared to other kernel types. In contrast, the Sigmoid kernel shows notably lower performance across all metrics, with an

F1-Score of 0.35, Precision of 0.53, Recall of 0.26, and Accuracy of 0.30. This suggests that the Sigmoid kernel may not be well-suited for this dataset, potentially due to its inability to effectively capture the underlying patterns. The Polynomial (Poly) kernel demonstrates competitive performance, particularly in terms of F1-Score and Precision, with values of 0.84 and 0.72 respectively. However, the Recall and Accuracy are slightly lower compared to the Linear kernel, indicating some trade-offs in performance. Finally, the Radial Basis Function (RBF) kernel performs consistently well across all metrics, with F1-Score, Precision, Recall, and Accuracy all achieving high values of 0.99. This suggests that the RBF kernel effectively captures the complex nonlinear relationships present in the dataset, resulting in robust performance.

Table 4.1: ***SVM Model Result Comparison on the KDD Dataset***

Metrics	**SVM-Kernels**			
	Linear	**Sigmoid**	**Poly**	**RBF**
F1-Score	0.87	0.35	0.84	0.99
Precision	0.78	0.53	0.72	0.99
Recall	0.99	0.26	1.0	0.99
Accuracy	0.80	0.30	0.72	0.99

In summary, the choice of SVM kernel significantly impacts the performance of the model on the KDD dataset. While the Linear and RBF kernels demonstrate strong overall performance, the Sigmoid kernel lags, indicating the importance of selecting an appropriate kernel type based on the dataset characteristics and modelling goals. Figure 4.1 presents a graphical representation of the performance of different SVM kernels, with their corresponding scores appended on the bars. The x-axis denotes the SVM kernels under comparison: Linear, Sigmoid, Poly, and RBF, enabling a direct comparison of their performance. Each kernel is distinctly separated along the x-axis. The y-axis represents the values of the performance metrics (F1-Score, Precision, Recall, Accuracy), providing a quantitative assessment of each SVM kernel's performance. The body of the bar chart comprises vertical bars corresponding to each SVM kernel. The height of each bar indicates the value of the respective performance metric for that kernel. For instance, the height of the bar for the Linear kernel at the F1-Score metric reflects its F1-Score value, and likewise for the other metrics. This visualization facilitates a comprehensive comparison of the effectiveness of each SVM kernel in network DDOS detection.

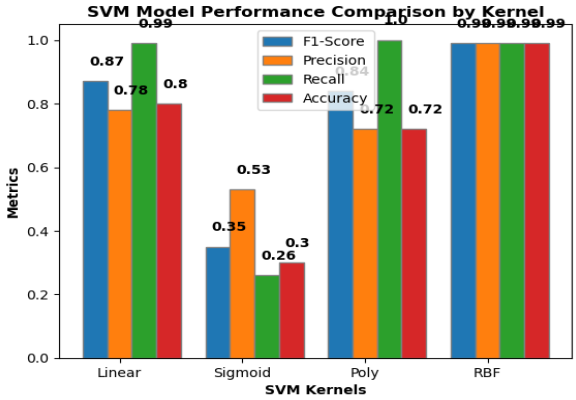

Figure 4.1: SVM Kernel Performance Graph

5.0 Conclusion

This study delved into the realm of network DDOS detection using the KDD-99 dataset obtained from Kaggle's machine-learning repositories. Leveraging machine learning algorithms, particularly the Support Vector Machine (SVM) algorithm, the research aimed to construct predictive models capable of identifying network intrusions. The research methodology adopted a structured approach comprising three fundamental phases: data preprocessing, classification, and performance evaluation. Data preprocessing involved refining and cleansing the raw data to ensure its suitability for subsequent analysis. This phase encompassed tasks such as handling missing values, normalizing data, and eliminating noise or outliers. Subsequently, SVM models were trained using various SVM kernels on the preprocessed training dataset. Finally, the performance of the SVM models was rigorously evaluated using an independent test dataset, employing metrics such as F1-score, precision, recall, and accuracy. The results and discussion revealed significant variations in model performance across different SVM kernels. The Linear and Radial Basis Function (RBF) kernels demonstrated robust performance across all metrics, while the Sigmoid kernel lagged. Specifically, the RBF kernel excelled in capturing complex nonlinear relationships present in the dataset, resulting in high-performance metrics across the board with precision, recall, f1-score, and accuracy of 99%. Conversely, the Sigmoid kernel exhibited subpar performance, indicating its limited suitability for the dataset. In conclusion, the choice of SVM kernel plays a crucial role in determining the efficacy of the model for network anomaly detection. While the Linear and RBF kernels showcased strong performance, the selection of an appropriate kernel type should be guided by the dataset characteristics and modeling objectives. This study sheds light on the importance of employing robust ML techniques for enhancing network security and intrusion detection capabilities. Further research could explore advanced methodologies and ensemble techniques to improve model performance and adaptability to evolving cyber threats.

Reference

Aggarwal, P., & Sharma, S. K. (2015). Analysis of KDD dataset attributes-class for intrusion detection. Procedia Computer Science, 57, 842-851.

Alharbi, Y., Alferaidi, A., Yadav, K., Dhiman, G., & Kautish, S. (2021). Denial-of-service attack detection over IPv6 network based on KNN algorithm. Wireless Communications and Mobile Computing, 2021, 1-6.

Gaye, B., Zhang, D., & Wulamu, A. (2021). Improvement of support vector machine algorithm in big data background. Mathematical Problems in Engineering, 2021, 1-9.

Ibor, A. E., & Epiphaniou, G. (2015). A hybrid mitigation technique for malicious network traffic based on active response. International Journal of Security and Its Applications, 9(4), 63-80.

Li, L., Yu, Y., Bai, S., Cheng, J., & Chen, X. (2018). Towards effective network intrusion detection: A hybrid model integrating gini index and GBDT with PSO. Journal of Sensors, 2018, 1-9.

Malik, A. J., & Khan, F. A. (2013). A hybrid technique using multi-objective particle swarm optimization and random forests for PROBE attack detection in a network. In 2013 IEEE International Conference on Systems, Man, and Cybernetics (pp. 2473-2478). IEEE.

Osanaiye, O., Choo, K. K. R., & Dlodlo, M. (2016). Distributed denial of service (DDoS) resilience in the cloud: Review and conceptual cloud DDoS mitigation framework. Journal of Network and Computer Applications, 67, 147-165.

Sharafaldin, I., Lashkari, A. H., Hakak, S., & Ghorbani, A. A. (2019). Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In 2019 International Carnahan Conference on Security Technology (ICCST) (pp. 1-8). IEEE.

Singh, A., & Gupta, B. B. (2022). Distributed denial-of-service (DDoS) attacks and defence mechanisms in various web-enabled computing platforms: issues, challenges, and future research directions. International Journal on Semantic Web and Information Systems (IJSWIS), 18(1), 1-43.

Yan, Q., Yu, F. R., Gong, Q., & Li, J. (2015). Software-defined networking (SDN) and distributed denial of service (DDoS) attacks in cloud computing environments: A survey, some research issues, and challenges. IEEE Communications Surveys & Tutorials, 18(1), 602-622.