A Comparative Analysis of Support Vector Machine and K-Nearest Neighbors Models for Network Attack Traffic Detection

Zhuoxi Han

School of Computer Science, Shanghai University, 200444 Shanghai, China

Abstract. With the continuous advancement of Internet technology, cybersecurity threats are growing more urgent as attack techniques become increasingly sophisticated. Conventional intrusion detection systems struggle to address these emerging threats because they depend heavily on predefined signatures and rules. This research centers on the use of advanced machine learning methods, particularly Support Vector Machines (SVM) and K-Nearest Neighbors (KNN), to improve the detection of network attack traffic. The UNSW-NB15 dataset, which includes various attack types and normal traffic patterns, is used to evaluate the performance of these two models. The results indicate that the SVM model excels in handling high-dimensional and intricate data, demonstrating its capability to tackle the complexity of modern cyber threats. On the other hand, KNN proves to be more efficient and straightforward when applied to less complex data structures. The outcomes of this study provide significant insights into enhancing cybersecurity systems, with recommendations for refining machine learning models to better address emerging threats. Moreover, the research highlights future directions to strengthen the resilience and precision of network intrusion detection systems, ensuring the development of more effective defenses against the ever-evolving landscape of cybersecurity risks.

1 Introduction

With the swift progression of Internet technology, the global informatization process has rapidly accelerated. The Internet has become an integral part of various areas of social life, such as personal communication, financial exchanges, industrial management, and national security, playing a key role in advancing globalization. Due to the advance in network technologies, the number of network users is growing rapidly, which leads to the generation of large network traffic data[1]. Cybersecurity is strongly linked to computers, networks, programs, and different forms of data, aiming to block unauthorized access and alterations [2]. The increase in network data is driven by the Internet of Things (IoT), cloud computing, and the vast number of connected devices[3]. As data traffic grows, so do cyberattacks, which have exponentially increased and pose a severe risk to cybersecurity. Cyberattacks can be

Corresponding author: 1014875532@qq.com

highly damaging, compromising an entire system by accessing, corrupting, or stealing data [4]. Traditional intrusion detection systems (IDS) primarily rely on rule-based matching and feature extraction, which are effective for identifying known threats. However, with the rapid evolution of attack strategies and the emergence of new types of threats, these traditional approaches are becoming less effective. Threats such as zero-day vulnerabilities, distributed denial of service (DDoS) attacks and advanced persistent threats (APT), and evolving phishing attacks make it challenging for IDS that rely on static rules and signatures to counteract them effectively. To overcome these limitations, numerous studies have explored deep learning techniques for threat detection. There have been several literature reviews [5-9] that have provided important insights in this area. Additionally, Berman et al. [5] provided the fundamentals of deep learning methods and their application background in attack detection, while Apruzzese et al. [6] focus on attack detection techniques in intrusion detection, malware analysis, and spam detection. Wickramasinghe et al. [7], on the other hand, provided a review of deep learning approaches secured using IoT technologies, clearly demonstrating the techniques used in various cyber attacks and detection. Aleesa et al. [8] reviewed intrusion detection systems based on deep learning in four major databases, using keywords such as "deep learning", "invasion" and "attack" to systematically sort out relevant literature and provide abundant resources for researchers. Emphasizing the importance of datasets in intrusion detection, Ferrag et al. [9] listed 35 well-known network datasets and divided them into seven categories, evaluating each category's accuracy and false positive rate through real traffic datasets, who reviewed deep learning-based intrusion detection systems across four major databases using terms like "deep learning," "invasion," and "attack," providing extensive resources for researchers. In the realm of traffic analysis for systems, widely used machine learning models. These models examine features of network traffic, to identify and classify potential attack patterns. SVM works by constructing a hyperplane in high-dimensional space to differentiate between normal and abnormal traffic, making it highly effective in high-dimensional datasets. KNN, on the other hand, classifies data by measuring the distance between samples, making it more suitable for simpler, lower-dimensional data scenarios.

2 Methodology

2.1 Network Anomaly Detection

This study utilizes the UNSW-NB 15 dataset and employs two classical machine learning models: KNN and SVM for experimental analysis and evaluation in network attack traffic detection. By training and assessing these models, This study demonstrates the respective advantages and disadvantages of various methods in anomaly detection tasks, while also providing references for the optimization of cybersecurity defense systems. Accuracy, precision, recall, F1 score, Receiver Operating Characteristic curve (ROC), and Area Under the Curve (AUC) metrics are employed to thoroughly analyze model functionality revealing SVM and KNN's different performances in network traffic anomaly detection. By comparing the performances of KNN and SVM in network attack traffic detection, this study reveals their respective strengths and weaknesses. Furthermore, it offers directions for future improvements in network intrusion detection systems. As network environments change constantly and attack methods evolve continually, exploring more efficient and accurate detection methods remains an important issue within the field of cybersecurity research.

The SVM distinguishes between normal and attack traffic by training a classifier. During the training phase, SVM learns from various traffic features, such as packet size, source address, and destination address, to construct an optimal hyperplane that separates normal

traffic from attack traffic. In the testing phase, new network traffic is mapped to this high-dimensional space, where SVM then determines whether it falls into the attack category.

The KNN model for network attack detection operates based on distance metrics to classify network traffic samples. It utilizes the similarity between traffic samples to detect and identify potential attack patterns. After converting network traffic data into feature vectors, KNN calculates the distances between the test sample and all training samples to identify the K nearest neighbors. Based on the class labels (normal traffic or attack traffic) of these neighbors, the KNN model predicts the category of the test sample as the label that appears most frequently among the neighbors.

The process involved reading the data, preprocessing, model training, prediction, evaluation, and visualization. The binary classification tasks were performed using SVM and K-Nearest Neighbors classifiers. The metrics employed to evaluate the proposed system include Accuracy, Precision, Recall, F1-Measure, ROC, and AUC [10].

2.2 Assessment Metrics for the KNN and SVM model

Accuracy assesses the proportion of instances that are classified correctly and serves as an indicator of the model's overall predictive effectiveness. The classification report presents precision, recall, and F1 score, providing a thorough assessment of the model's performance for each individual category. The confusion matrix showcases the comparison between actual and predicted labels, helping to analyze the model's behavior across different classes.

3 Experimental result and Analysis from Different Models

3.1 Experimental result

Table 1. SVM model performance

Label	Precision	Recall	F1-score	Support
0	0.36	0.87	0.51	56000
1	0.82	0.28	0.41	119341

As shown in Table 1, the whole accuracy is 0.47, based on 175,341 instances. The precision averaged across all classes is 0.59, with a recall of 0.58 and an F1 measure of 0.46. The class-weighted precision is 0.68, along with a recall of 0.47 and an F1 measure of 0.45.

3.2 SVM Model Analysis

Accuracy represents the proportion of correctly classified samples and is utilized to assess the overall predictive performance of the model. The classification report features precision, recall, and F1 score, offering a detailed evaluation of classification efficiency for each category. The confusion matrix displays the alignment between true labels and predicted labels, assisting in the analysis of model performance across different classes. The ROC plot demonstrates the balance between the rate of false positives and true positives across different threshold levels, while the AUC score represents the model's overall classification effectiveness, with a higher AUC indicating stronger performance.

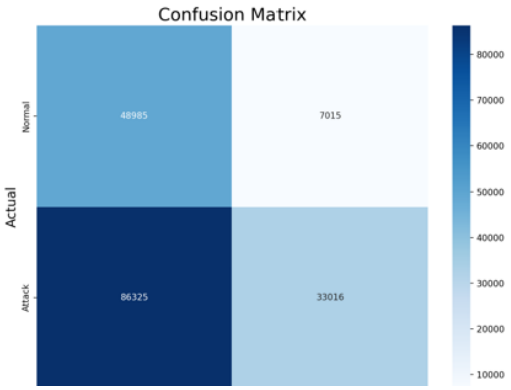

Fig. 1. Confusion matrix for SVM **(Picture credit : Original)**

Fig. 2. ROC curve for SVM (Picture credit : Original)

As shown in Figure 1, the training outcomes of the SVM system demonstrate these features: The accuracy is 0.577, indicating that the model performs similarly to KNN, with about 57.7% of samples being correctly classified. The positive predictive value is 0.87, indicating that when the model classifies a sample as normal, there is an 87% likelihood that it is correct. However, the relatively high rate of false positives suggests that SVM slightly underperforms compared to KNN when it comes to classifying normal samples. The recall stands at 0.36, meaning only 36% of the samples that are truly normal are identified as such, indicating a high rate of missed attack detections. According to the confusion matrix, there are 48,985 true positives (TP), 7,015 false positives (FP), 86,325 false negatives (FN), and 33,016 true negatives (TN). This indicates that while SVM performs moderately in classifying normal samples, it leaves much room for improvement in detecting attacks. As depicted in Figure 2, the AUC value is 0.73, showing that the model has solid classification capabilities and can effectively differentiate between positive and negative samples. The contour is positioned near the upper-left corner, suggesting a significant boost in the true positive rate with a low false positive rate, indicating stronger performance in identifying positive cases. SVM's performance surpasses that of KNN, likely due to its ability to find the optimal separating hyperplane in high-dimensional spaces, giving it an edge when handling complex datasets.

3.3 KNN Model Analysis Network intrusion data

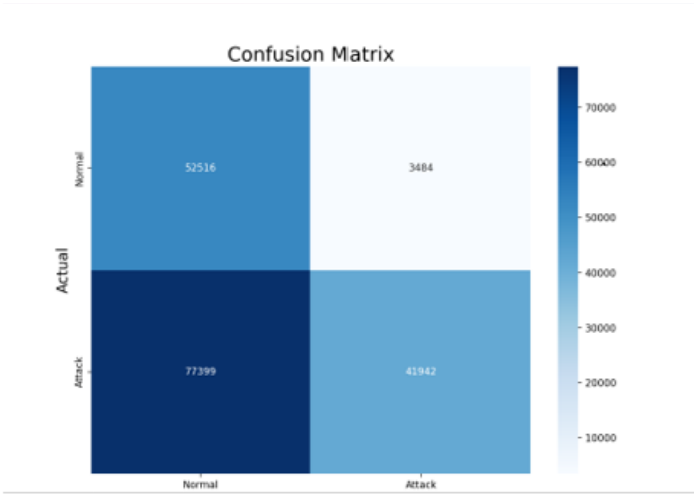

Fig. 3. Confusion matrix for KNN (Picture credit : Original)

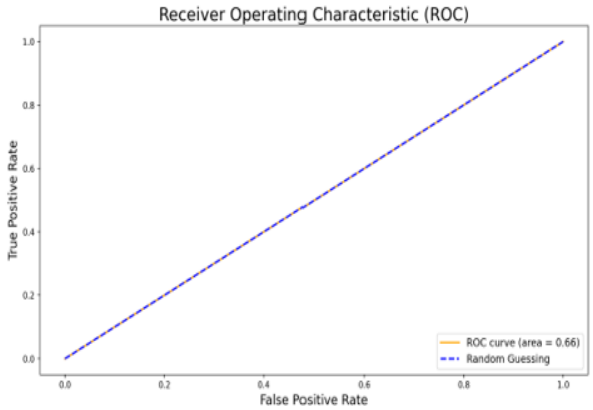

Fig. 4. ROC curve for KNN (Picture credit : Original)

As depicted in Figure 3, the training results of the KNN system reveal these characteristics: The accuracy is 0.577, showing that the model performs moderately, with around 57.7% of samples being classified correctly. The positive predictive value is 0.93, indicating that when the model classifies a sample as normal, it is correct 93% of the time. The relatively low number of false positives highlights KNN's high accuracy in classifying normal samples. The recall is 0.40, which indicates that only 40% of the samples that are truly normal are recognized as such, suggesting that the model misses some attack samples. The confusion matrix shows there are 52,516 TP, 3,484 FP, 77,399 FN, and 41,942 true TN. While KNN is effective at classifying normal samples, there is still room for improvement in detecting attack samples. As shown in Figure 4, the KNN model exhibits the following characteristics during training: The accuracy is 0.577, meaning the model performs moderately overall, with about 57.7% of samples correctly classified. The precision of 0.93 implies that when the model predicts a sample as normal, it has a 93% likelihood of being correct. The low number of false positives indicates that KNN is highly precise in identifying normal samples. The recall is 0.40, meaning only 40% of the samples that are genuinely normal are correctly

identified, implying the model misses a significant number of attack samples. According to the confusion matrix, there are 52,516 TP, 3,484 FP, 77,399 FN, and 41,942 TN. This indicates that while KNN is strong at classifying normal samples, it still struggles with identifying attacks. The AUC score is 0.66, suggesting that the model's classification performance is moderate, with weaker ability in distinguishing between positive and negative cases. The curve is positioned further from the upper-left corner, indicating that the model only attains a minor enhancement in the true positive rate at lower false positive rates. Overall, KNN's classification accuracy may be influenced by noise and the data distribution, resulting in less satisfactory outcomes than expected.

3.4 Comparison of KNN and SVM

Contrast the ROC curves and AUC metrics of KNN and SVM leads to the following conclusions:The AUC metric of SVM (0.73) is significantly greater than that of KNN (0.66), indicating superior classification performance by SVM.KNN shows limited improvement in the true positive rate at low false positive rates, demonstrating weaker recognition ability, whereas SVM performs better under the same conditions, effectively identifying positive samples.Overall, SVM excels in handling complex data, making it more suitable for scenarios requiring high classification accuracy, whereas KNN's performance on this dataset is relatively poorer.

4 Conclusion

This research evaluates the effectiveness of KNN and SVM models in detecting network attack traffic, highlighting their differences. The experimental findings show that SVM exhibits better generalization when working with high-dimensional and complex datasets, excelling in differentiating between normal and malicious traffic and achieving strong results across various performance metrics. In contrast, KNN performs well in simpler data scenarios but struggles with high-dimensional and large-scale datasets, often resulting in higher false positive rates when handling imbalanced data. Future research should prioritize the following: improving the real-time detection capabilities and efficiency of these systems to manage the increasing volume of network data; refining algorithms to better balance false positive and false negative rates, particularly when dealing with imbalanced datasets; and creating more efficient detection methods suited for the IoT environment, addressing its inherent complexity and diversity. These research directions will serve a critical purpose in advancing network traffic detection technologies and their real-world applications.

References

1. W.A. Ali , K.N. Manasa , M. Bendechache, A review of current machine learning approaches for anomaly detection in network traffic. Journal of Telecommunications and the Digital Economy. **8(4)**, 64-95 (2020)
2. S. Aftergood , Cybersecurity: the cold war online. Nature. **547**, 30-31 (2017)
3. M. Belgiu, L. Dragut, Random forest in remote sensing: A review of applications and future directions. ISPRS journal of photogrammetry and remote sensing. **114**, 24-31 (2016)
4. M.-A. Al-Garadi, A. Mohamed , A. Al-Ali ,X. Du , and M. Guizani , A survey of machine and deep learning methods for internet of things (iot) security. IEEE communications surveys & tutorials. **22**, 1646-1685 (2020)

5. D. Berman , A. Buczak, J. Chavis, and C. Corbett, A survey of deep learning methods for cyber security, Information. **10.4**, 122 (2019)
6. G. Apruzzese, M. Colajanni, L. Ferretti, A. Guido, and M. Marchetti, On the effectiveness of machine and deep learning for cyber security, Proceedings of 2018 10th International Conference on Cyber Conflict (CyCon), IEEE, (2018), 371-390
7. C. S. Wickramasinghe, D.L. Marino, K. Amarasinghe, and M. Manic, Generalization of deep learning for cyber-physical system security: a survey. In IECON 2018-44th Annual Conference of the IEEE Industrial Electronics Society, IEEE, 2018, 745–751
8. A. Aleesa , B. Zaidan, A. Zaidan, and N.M. Sahar, Review of intrusion detection systems based on deep learning techniques: coherent taxonomy, challenges, motivations, recommendations, substantial analysis and future directions. Neural Computing and Applications. **32**, 9827-9858(2020)
9. M.A. Ferrag, L. Maglaras, S. Moschoyiannis, and H. Janicke, Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study. Journal of Information Security and Applications. **50**, 102419(2020)
10. E.G. Dada, J.S. Bassi, H. Chiroma, A.O. Adetunmbi, O.E. Ajibuwa, Machine learning for email spam filtering: review, approaches and open research problems. Heliyon. **5**, 6(2019)