Intrusion Detection Using Random Forest Classifier: A Machine Learning Approach with Advanced Metrics

Humza Rana

Department of Computer Science, Bahauddin Zakariya University, Multan, Pakistan

Abstract: To protect the network assets, online business, and user banking information from hacking or fraud. Advanced and multi-layered security is needed to control these issues. In modern days, the use of artificial intelligence in our daily life activities is increasing day by day. Different type of machine learning algorithms that comes from AI help us to solve our problems. In this paper, the machine learning algorithm, random forest classifier, is proposed to improve the intrusion detection system. The WSN dataset used in this paper is multiclass. After the proposed model performance, a comparison is presented between different machine learning algorithms. The proposed model shows an accuracy of 0.995%, which shows the best results after comparison.

Keywords: Artificial Intelligence, Machine Learning, Intrusion Detection, SMOTE, etc.

Email: Humza.Rana99@gmail.com

1. Introduction

The quantity of computational devices is increasing at a high rate. Due to the increase in devices, the usage of the internet has also become very common nowadays. Now it needs to protect the online activity from misuse, hacking, and others stealing, which is why IDS is required. The IDS helps protect from malicious acts or any behavior that is harmful to the users. The IDS is an application or device that plays a role in protecting from malicious acts or attackers [1]. The use of the internet in today's world has become so important because every user activity now gets online; every business, transaction, banking system, and education system is now online. There is an increased risk that attackers will hack this most critical information to demand money. The IDS examines the network traffic to detect any type of malware or malicious movement in traffic. IDS are in different types that identify the malicious acts in traffic by matching the existing intrusion with new traffic with these types of intrusion. The other type determines whether the traffic behavior is normal or not [2]. The main objective of this paper is to provide the IDS with the best ML algorithms that will classify different types of attacks.

1.1 Contribution

One of the commonly used machine learning algorithms for detecting intrusions within network traffic is the Random Forest classifier. Unlike some primitive classifiers, which cater for only two classes like “attack” or “normal,” Random Forest is built for multiclass classification. This

indicates its ability to differentiate several types of network intrusions as opposed to just telling apart normal and malicious traffic. For the purposes of this research, the authors selected a WSN dataset to train and test the model. Wireless sensor networks are utilized and needed in a variety of areas, such as the environment and smart cities. Protection of these networks is very important, as they are often deployed in sensitive or exposed areas. One important aspect of an intrusion detection dataset is the challenge of class imbalance. An imbalance of classes occurs due to some types of intrusions being much rarer than others. Take, for example, network traffic: while some attacks may be less frequent, the amount of network traffic is quite abundant. This disproportionality in the sample data set can lead to poor machine learning algorithms, especially when there is a disproportionate bias to the majority class. In this case, the study utilized the SMOTE approach. This is the system SMOTE, which stands for "Synthetic Minority Over-sampling Technique", used to balance data sets. Duplicating samples is not an option; in fact, SMOTE uses interpolation of underrepresented samples to derive new ones, which in turn balances out the class inequalities. This, in turn, allows the model to better learn the representation of infrequently occurring attack types, which leads to a higher detection of such attacks. The evaluation done by the researchers on the performance of the Random Forest classifier included more sophisticated metrics for evaluation other than accuracy. Relying on accuracy is dangerous, particularly with imbalanced datasets. Rather, metrics like precision, recall, F1-score, and others paint a fuller picture of how a model performs with every class, especially with infrequent attacks. The performance of the Random Forest model was evaluated against other machine learning models. This is done to validate the effectiveness of the proposed approach concerning other well-known classifiers like Support Vector Machines (SVM), Decision Trees, and even Neural Networks. Results indicated that the Random Forest classifier, combined with SMOTE and more sophisticated metrics, was more efficient and accurate in detecting different forms of network intrusions.

1.2 Problem

While the signature-based Intrusion Detection System (IDS), which compares network traffic with known attack signatures, is useful to identify already known threats, it is riddled with obvious drawbacks. By comparing patterns observed above with a database of known attacks, such systems can predict whether malcontent is afoot… but they will never even have seen the unknown or zero-day threat. Attacks come in all shapes and sizes — a gap that allows attackers

to avoid detection by using entirely new techniques or slight variants of known attack patterns. It is an even bigger problem with large networks, as much data needs to be inspected by an IDS that needs to be highly optimized for speed and scalability. However, both require the network traffic to be processed at line-rate, which is a tough challenge because even IDS requirements are high, and an IDS device may miss real-time threats due to a lack of efficient architecture and processing, resulting in delayed detection or more false positives. Older signature-based IDS technologies are additionally resource-intensive as they demand ongoing signature updates and still fail to identify high-level malicious activities like metamorphic malware, encoded threats, or multi-stage attacks. Tools such as these will provide protection to a degree, but they do not detect advanced or obfuscated threats; Firewalls block untrusted access based on rules, and scanners are able to identify known vulnerabilities or malware at a point in time with no holistic behavioral monitoring. To overcome these limitations, Many of the newer IDS system designs now incorporate machine learning, anomaly detection, and behavioral analytics, which enable the examination of passing traffic for not only known but also emerging attack types, while still scaling in performance even for large-scale operations that run high on Network throughputs.

2. Related Work

Tahri et al. 2022 [1] implement three ML algorithms on two different datasets to detect the intrusion. The datasets USNW, NB-15, and NSL-KDD were used in this experiment. Three ML algorithms include the SVM, KNN, and NB. After an experiment, SVM performs better than the two algorithms. The 97.7% accuracy attained by SM in intrusion detection. Palimote et al. 2021 [3] proposed ML algorithms for detecting network intrusion. The two algorithms, which are Random Forest and SVM, were used. The US Air Force TCP/IP network dataset was used in this experiment. The dataset contains two classes. After the experiment, the models generated 99.7% and 94.6% accuracy performance in network intrusion detection. Khedkar et al. 2023 [4] proposed machine learning algorithms that use SMOTE in them. The SMOTE applies to the K-Means algorithms. This proposed model was applied to the NSL-KDD dataset. After comparison with different algorithms, the RF gives 95.8% accuracy in the detection of network intrusion.

Khaldi et al. 2024 [5] implement different ML algorithms to detect the intrusion in medical-based health care applications for IoMT. The NSL-KDD dataset used in this experiment contains 41 features. The feature selection techniques also apply in this experiment. The PCC and backward elimination wrapper were used on features. After an experiment with different ML

algorithms, the SVM performance increased from 43% to 85% accuracy after feature selection and cross-validation. Phatak et al. 2020 [6] implement two ML algorithms, DT and KNN. The ANOVA feature selection applies to these algorithms to evaluate the performance in intrusion detection. The NSL-KDD dataset was used in this experiment. After an experiment, the decision tree performed with 99.15% accuracy compared to the KNN. Ahmad et al. 2022 [7] proposed AdaBoost on the DT classifier to detect the intrusion. The AdaBoost is implemented using correlation values among features. The USNW-NB15 dataset was used in this experiment. After comparison with different ML algorithms, the AdaBoost DT gives 99.3% accuracy performance in the detection of network intrusion.

3. Methodology

The following diagram explains the data preprocessing of the dataset features, the training of the proposed model, and the prediction as given below.

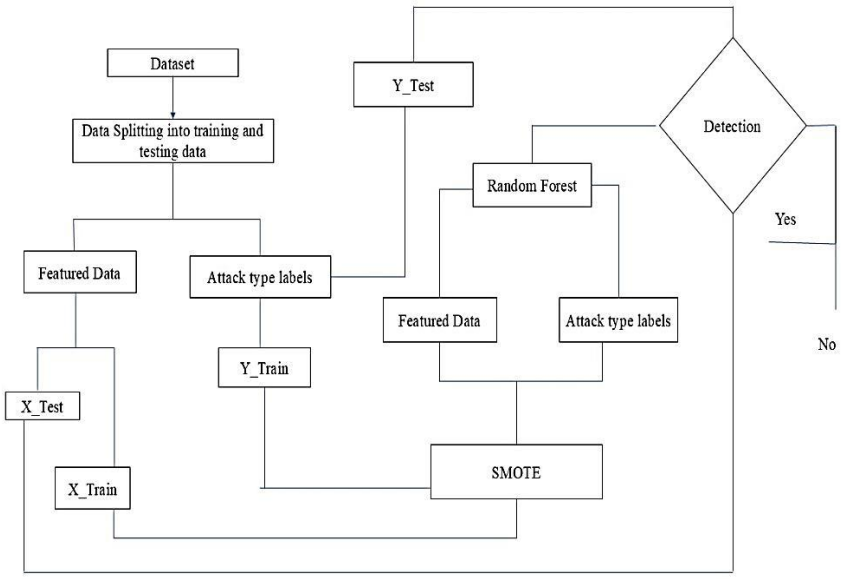

Figure 1 Proposed Model with Data Preprocessing Steps

Figure 1 presents a machine learning flow for detection-based attacks beginning with a data set that contains features (input variables) and corresponding labels of attack types. The data set is broken into training and testing sets, in which the features are separated from their labels. Preprocessing of the training features X-train and labels Y-Train by SMOTE will balance the data set to avoid any bias toward the majority class or classes. This balanced training set will then be used to train a Random Forest classifier. After it has been trained, the model makes

predictions on the test features X_Test and compares them to the actual test labels Y_test. If the output evaluated at the detection stage reflects that the system is able to detect attacks, then that would be termed successful detection; otherwise, there would be a need for adjustments or even retraining of the whole system. The workflow balances training and avoids any sort of data leakage since SMOTE is applied only to the training data set and enables a feedback loop for enhancing detection performance.

4. Results

A) Wireless Sensor Network

The Wireless sensor network dataset contains 350,000 instances with five malicious types. It contains the normal, gray hole, black hole, TDMA, and flooding attack types.

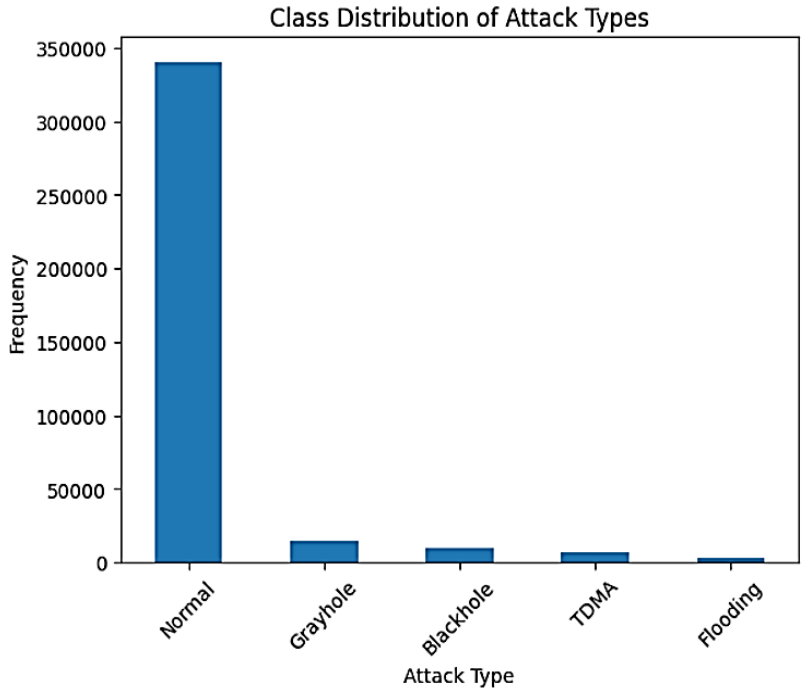

Figure 2: WSN Dataset Instances per class

Figure 2 shows the count of various types of activities on a network. These types of activity alongside the imbalance between different categories of activity in the dataset. “Normal” traffic types far exceed the rest, normal traffic clocking over 300,000, while the various attack types, gray hole, black hole TDMA, and flooding, only have counts in the thousands or hundreds. They are nearly non-existent next to the “Normal” bar, and the ratio between the two is drastic. This is

quite common in cybersecurity, where there is a tremendous amount of authentic network traffic and very little malicious activity. This extreme imbalance in class distributions could severely impede the creation of effective machine learning models, which would find it difficult to accurately detect the minority attack classes.

Table 1 Proposed Model comparison with different ML algorithms

Models	Accuracy	Precision	Recall	F1-Score
Proposed Model	0.995%	0.995%	0.995%	0.995%
Decision Tree	0.992%	0.992%	0.992%	0.992%
Naive Bayes	0.815%	0.932%	0.8192%	0.865%
Support Vector Machine	0.822%	0.931%	0.823%	0.861%
KNN	0.966%	0.971%	0.966%	0.970%

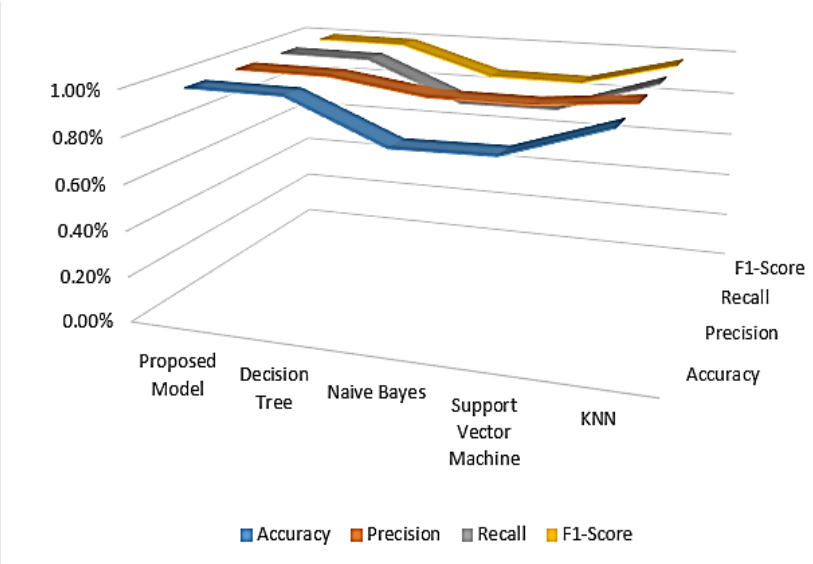

Figure 2: Proposed Model Comparison with different Metrics

Figure 2 explains a 3D line chart comparing the number of misclassifications in various Machine learning models (View Large). 4-Metrics_Comparison: This image displays the performance for different machine learning models across four metrics — Accuracy, Precision, Recall, and F1-Score. Proposed Model Decision Tree Naive Bayes Support Vector Machine KNN, wherein the X-axis represents different machine learning models, while the Y-axis represents the performance scores in percentages. The legend at the bottom should reveal that blue = the

Accuracy, orange = Precision, gray = Recall, and yellow=F1-Score. Looking at the visualization, it is easy to see that in general, for all metrics (Accuracy, F1-Score…), "Proposed Model" has, most of the time, the highest dots from its lines, which are almost always on top, above other models, especially for Accuracy and F1-Score. While models such as "Naive Bayes" and "Support Vector Machine" show that the performance drops slightly, KNN seems to have more variety in performance.

Table 2 Proposed Model Comparisons with Advanced Metrics

Models	Log Loss	Matthews Correlation Coefficient	Cohen's Kappa Score
Proposed Model	0.026	0.975%	0.975%
Decision Tree	0.283	0.951%	0.951%
Naive Bayes	1.670	0.468%	0.407%
Support Vector Machine	0.644	0.485%	0.418%
KNN	0.536	0.835%	0.828%

Table 2 presents the proposed model's log loss, Matthews coefficient, and Cohen's kappa score with different machine learning algorithms.

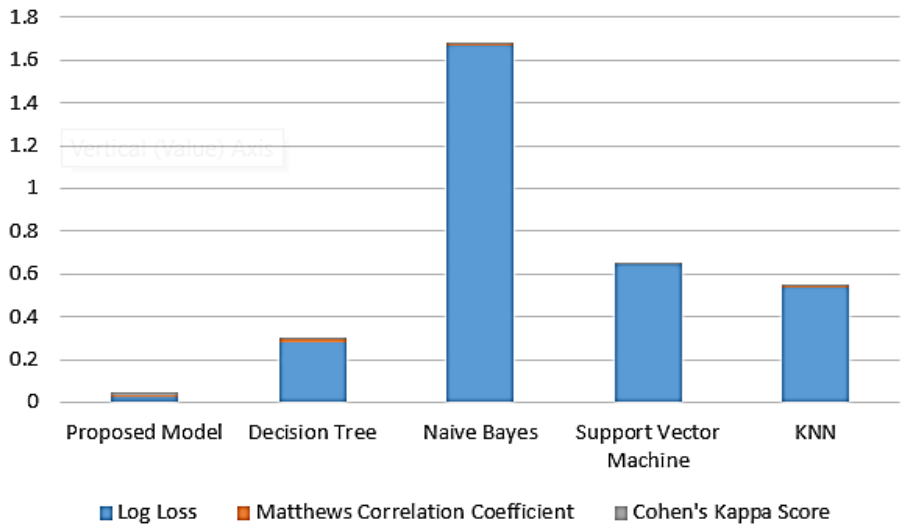

Figure 3: Proposed Model Comparison with Advanced Metrics

Figure 3 explains the proposed model comparison chart with different machine learning algorithms. The comparison states that the proposed model achieves the lowest log loss.

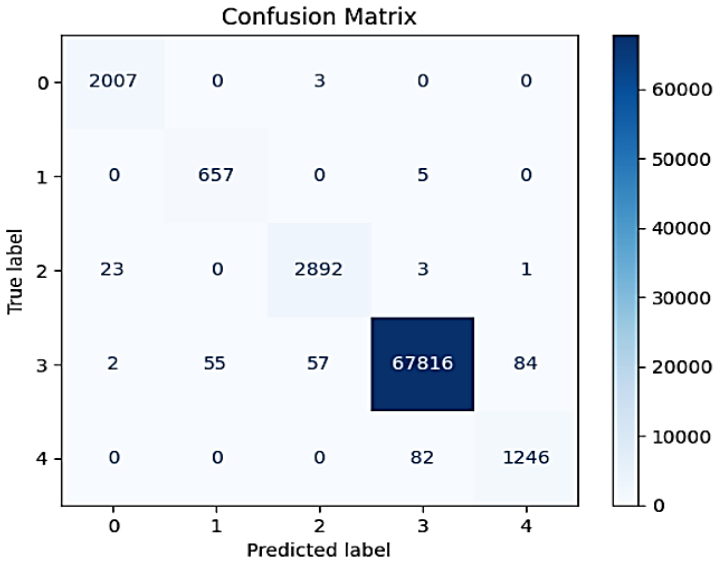

Figure 4: Confusion Matrix from the Proposed Model on Prediction

Figure 4 explains the confusion matrix for the proposed model's performance on a multiclass dataset with approximately five malicious class predictions.

5. Discussion

After exploring and experimenting with the results from the proposed model, it is clear that Random Forest performs better than other machine learning algorithms. It achieves 0.995% accuracy on a multiclass dataset with five classes, yielding the best performance. The confusion matrix presents per-class predictions by the proposed model. The SMOTE helps to select suitable features from the dataset that increase the model performance and time.

6. Conclusion

In this paper, the random forest algorithms are proposed and compared with the different machine learning algorithms. Using SMOTE to balance the WSN dataset, which is multiclass and contains five types of attacks. The proposed model achieves 0.995% accuracy in intrusion detection. The model performance is compared with different ML algorithms to determine whether the proposed model is more efficient than the other models.

7. Future

From the best performance on the WSN dataset, the next step is to apply the proposed model to the Windows APIs dataset in a multiclass setting to check model performance and further evaluate the model on the APIs dataset, which includes malicious and different attack types.

References

[1]. R. Tahri, Y. Balouki, A. Jarrar, and A. Lasbahani, “Intrusion Detection System Using Machine Learning Algorithms,” *ITM Web Conf.*, vol. 46, p. 02003, 2022, doi: 10.1051/itmconf/20224602003.

[2]. E. E. Abdallah, W. Eleisah, and A. F. Otoom, “Intrusion Detection Systems using Supervised Machine Learning Techniques: A survey,” *Procedia Comput. Sci.*, vol. 201, no. C, pp. 205–212, 2022, doi: 10.1016/j.procs.2022.03.029.

[3]. J. Palimote, L. Atu, and E. Osuigbo, “A Model to Detect Network Intrusion Using Machine Learning,” vol. 8, no. 1, pp. 521–527, 2021.

[4]. S. Khedkar and M. Babhulgaonkar, “Intrusion Detection System using K-Means Clustering and SMOTE,” *Int. Res. J. Eng. Technol.*, pp. 577–585, 2023, [Online]. Available: www.irjet.net

[5]. M. Khaldi, N. Mahammed, M. A. Lahmar, and F. D. Daouadji, “An Intrusion Detection System for Healthcare Applications using Machine Learning,” *CEUR Workshop Proc.*, vol. 3694, pp. 94–101, 2024.

[6]. Ashwini Pathak and Sakshi Pathak, “Study on Decision Tree and KNN Algorithm for Intrusion Detection System,” *Int. J. Eng. Res.*, vol. V9, no. 05, pp. 376–381, 2020, doi: 10.17577/ijertv9is050303.

[7]. I. Ahmad, Q. E. U. Haq, M. Imran, M. O. Alassafi, and R. A. Alghamdi, “An Efficient Network Intrusion Detection and Classification System,” *Mathematics*, vol. 10, no. 3, pp. 1–15, 2022, doi: 10.3390/math10030530.