Random Forest Machine Learning for Enhanced Network Intrusion Detection

B. SANTOSH KUMAR

Associate Professor, Department of MCA, Wesley PG College, Secunderabad, India.

Abstract - With the increasing sophistication of cyber threats, effective network intrusion detection has become paramount in ensuring the security of computer systems and sensitive data. This paper presents a comprehensive investigation into the application of Random Forest machine learning for network intrusion detection. The Random Forest algorithm is chosen for its ability to handle diverse and complex network traffic patterns, providing a robust solution to discern between normal and malicious activities. The study begins by preprocessing network traffic data, including feature engineering techniques, to optimize the input for the Random Forest model. Subsequently, a Random Forest Classifier is trained on labeled datasets encompassing both benign and intrusive network behaviors. The model is fine-tuned through cross-validation to optimize hyperparameters, ensuring adaptability to varying network conditions. Evaluation of the model's performance includes key metrics such as accuracy, precision, recall, and F1 score, offering insights into its ability to effectively detect and classify network intrusions.

Keywords: Network intrusion detection, Random Forest, Machine learning, Cyber threats, Security of computer systems, Feature engineering

I. INTRODUCTION

The field of cybersecurity places great importance on the task of network intrusion detection, which serves to protect computer networks from unauthorized access and destructive actions [1]. The proliferation of network interconnection and the widespread presence of digital technologies have significantly increased the susceptibility of networks to a wide range of cyber-attacks. The range of threats include several types of assaults, such as virus invasions, denial-of-service (DoS) attacks, and unauthorized access attempts, among others [2]. The ability to promptly identify and address these unauthorized accesses has become crucial in safeguarding the integrity and security of network infrastructures.

In light of the dynamic nature of cybersecurity threats, the difficulties encountered in accurately identifying and promptly addressing network breaches have become more pronounced [3]. The ability of traditional rule-based approaches and signature-based systems to effectively respond to the constantly evolving landscape of cyber threats is often challenged [4]. The ever-changing nature of assaults and the development of new intrusion tactics consistently avoid pre-established rules, hence diminishing the efficacy of traditional detection systems. Moreover, the considerable quantity and intricate nature of network data intensify the challenge of identifying anomalous patterns that signify intrusions among the extensive array of authorized network operations [5].

The use of machine learning has been more prominent as an effective method to address the growing complexities linked to network intrusion detection [6]. The autonomous ability of machine learning models to identify patterns and abnormalities within intricate datasets makes them a potential approach to address the ever-changing nature of cyber threats [7]. These models use network traffic data analysis to detect little variations from typical activity, therefore facilitating the identification of abnormal behaviors that indicate possible intrusions.

The use of machine learning models in the area of network intrusion detection takes advantage of such models' ability to learn and adapt based on extensive amounts of network data [8]. Supervised learning approaches, such as Support Vector Machines (SVM), Random Forests, and Neural Networks, have the power to extrapolate patterns obtained from previous data in order to identify and categorize prospective intrusions in real time. This is an advantage over unsupervised learning techniques, which do not have this capability. Unsupervised learning methods, such as clustering and anomaly detection algorithms, are very important components in the process of determining whether or not network data exhibits deviations from the expected patterns of behavior. Network intrusion detection systems aim to improve their effectiveness in rapidly detecting and mitigating cyber attacks and strengthening the resilience of network infrastructures against possible intrusions by using various machine learning approaches.

The present research explores the use of machine learning techniques to strengthen network intrusion detection systems, with the goal of improving the effectiveness of cybersecurity measures in protecting vital digital resources from emerging cyber risks.

II. LITERATURE

Kazi Abu Taher et al [9] demonstrated the use of Artificial Neural Network (ANN) based machine learning, coupled with wrapper feature selection, provides superior performance compared to the support vector machine (SVM) approach in the classification of network traffic. The performance evaluation involves the use of the NSL-KDD dataset for the purpose of classifying network traffic via the application of supervised machine learning methods, namely Support Vector Machines (SVM) and Artificial Neural Networks (ANN). A comparative analysis reveals that the suggested model has higher efficiency in terms of intrusion detection success rate when compared to other current models.

Razan Abdulhammed et al [10] implemented two approaches to decrease the dimensionality of features. To begin with, an Auto-Encoder (AE), which is a prominent example of deep learning methodologies, is used for the purpose of reducing dimensionality. Furthermore, Principle Component Analysis (PCA) is used as a supplementary method. Following this, the extracted low-dimensional features from both approaches are utilized in the construction of various classifiers, such as Random Forest (RF), Bayesian Network, Linear Discriminant Analysis (LDA), and Quadratic Discriminant Analysis (QDA), to facilitate the development of an Intrusion Detection System (IDS). The study conducted experiments to investigate the impact of using low-dimensional features in both binary and multi-class classification situations. The results of the research indicate that including these features leads to better performance measures, including increased Detection Rate (DR), enhanced F-Measure, decreased False Alarm Rate (FAR), and higher Accuracy.

Preeti Mishra et al [11] presented a comprehensive examination and evaluation of several machine learning methodologies in order to identify the underlying factors contributing to the challenges encountered in the detection of invasive activities. The categorization and mapping of assault characteristics are supplied in correspondence to each attack. This paper examines the challenges associated with the detection of low-frequency assaults using network attack datasets. It also proposes realistic strategies for enhancing the detection capabilities in this context. The detection capabilities of different machine learning algorithms have been thoroughly researched and evaluated in relation to their ability to identify assaults across many categories. The limitations pertaining to each group are also addressed. The study incorporates a range of data mining technologies for machine learning. In conclusion, this study offers potential avenues for further investigation in the realm of attack detection via the use of machine learning methodologies.

M Elif KarsligEI et al [12] presented a novel semi-supervised anomaly detection system that utilizes the k-means algorithm for the purpose of identifying network threats. During the training phase, the system employs the k-means method to first partition normal data samples into separate clusters. Following this, in order to distinguish between normal and abnormal samples, a threshold value is computed using distances from the centers of the clusters. This is done by using a validation dataset. Anomalies detected when fresh samples demonstrate distances that exceed the threshold value from the cores of the clusters. In order to assess the effectiveness of the suggested methodology, the NSL-KDD dataset, which offers labeled network connection traces, was used for thorough testing objectives.

Mohammad Almseidin et al [13] presented a comprehensive analysis of many experiments conducted to evaluate the performance of different machine learning classifiers using the KDD intrusion dataset. The computation of several performance measures was successful in order to assess the chosen classifiers. The primary emphasis was placed on evaluating the performance metrics of false negatives and false positives with the objective of improving the detection rate of the intrusion detection system. The conducted trials provided evidence that the decision table classifier exhibited the lowest occurrence of false negatives, whilst the random forest classifier had the greatest mean accuracy rate.

Chie-Hong Lee et al [14] implemented the equality constrained-optimization-based extreme learning machine as a methodology for network intrusion detection. This study presents a unique approach to adaptively incrementally train in order to discover the optimal number of hidden neurons. The study introduces optimization criteria and a binary search strategy for dynamically augmenting hidden neurons. The efficacy of the suggested technique is assessed by its implementation in the field of network intrusion detection. The effectiveness of this strategy in developing models with high attack detection rates and fast learning speeds has been supported by empirical evidence.

Manjula C. Belavagi et al [15] presented the construction of classification and prediction models for intrusion detection via the use of machine learning classification methods, namely Logistic Regression, Gaussian Naive Bayes, Support Vector Machine, and Random Forest. The algorithms undergo testing using the NSL-KDD dataset. The experimental findings demonstrate that the Random Forest Classifier has superior performance compared to other approaches in accurately distinguishing between regular data flow and malicious attacks.

III. PROPOSED MODEL

The use of machine learning in network intrusion detection is the utilization of computational models to autonomously detect and address anomalous activity or possible security risks inside a computer network. The system utilizes machine learning algorithms to acquire knowledge of patterns and behaviors derived from previous network data, enabling it to differentiate between regular network traffic and potentially harmful or suspicious actions.

The Random Forest (RF) classifier is a machine learning ensemble technique often used for diverse classification purposes, such as network intrusion detection. The architectural design of a Random Forest classifier consists of several decision trees, with each tree being trained on a specific subset of the dataset. The final classification is established by a process of majority voting or averaging the predictions made by each individual tree. The proposed architecture is shown in figure 1.

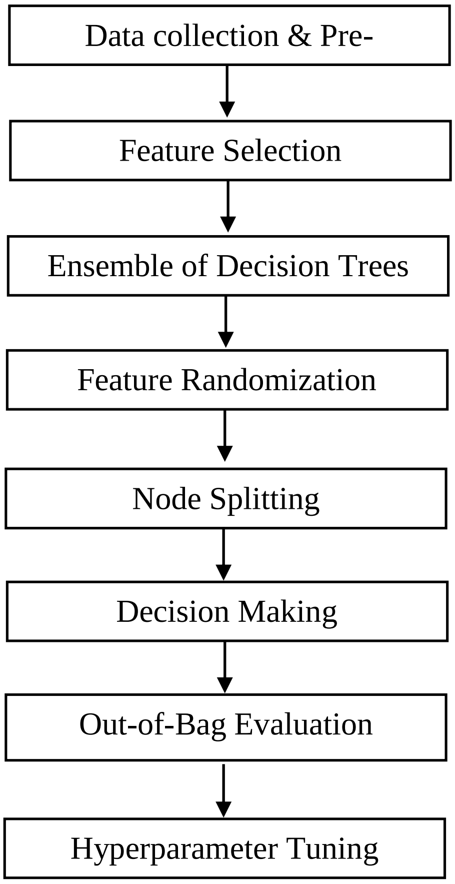


Figure 1: Proposed model Architecture

The following discourse presents a comprehensive elucidation of the architectural framework pertaining to network intrusion detection.

- **Data Collection and Preprocessing:**

The process of selecting a dataset for analysis Please choose an appropriate dataset for the purpose of network intrusion detection. Data preprocessing involves the cleaning and preprocessing of the dataset. In order to address the issue of missing data, it is necessary to use appropriate techniques such as imputation or deletion. Additionally, categorical variables should be encoded using suitable methods such as one-hot encoding or label encoding. Furthermore, numerical characteristics should be normalized or scaled to ensure comparability and prevent any undue influence on the analysis.

- **Feature Selection:**

Determine the relevant characteristics for the purpose of intrusion detection. In order to identify the most useful characteristics, it is recommended to employ several strategies, such correlation analysis, recursive feature removal, or drawing upon domain experience.

- **Ensemble of Decision Trees:**

The ensemble of decision trees is a commonly used technique in machine learning and data mining. It involves combining many decision trees to make predictions or classify data. This approach is known for its ability to handle complex and non The Random Forest classifier is comprised of a collection of decision trees. Each tree is built separately and then trained using a randomly selected sample of the training data. The procedure referred to as bagging, short for Bootstrap Aggregating, involves the creation of several subsets, or bags, by randomly sampling with replacement from the original dataset.

- **Feature Randomization:**

The concept of feature randomization refers to the technique of introducing randomness into the selection or arrangement of features in a given context. In order to promote variety within the tree population, a stochastic selection process is used wherein a subset of characteristics is randomly chosen for consideration during the splitting process at each node of the decision tree. This practice aids in mitigating the risk of individual trees being too specialized and overfitting to certain characteristics within the dataset.

- **Node Splitting:**

The concept of node splitting refers to a technique used in several fields, particularly in computer science and data structures. At every node inside the decision tree, a feature is chosen using a criterion such as Gini impurity or information gain. Subsequently, the node is divided into child nodes depending on this selection. The aforementioned procedure is iteratively performed until a termination condition is satisfied, such as attaining a certain level of depth or achieving a specified minimum quantity of samples inside a terminal node.

- **Decision Making:**

The process of decision making is a fundamental aspect of human behavior and cognitive processes. The ultimate determination of the classification outcome for a novel instance is achieved by consolidating the individual predictions made by all the trees. In the context of a classification job, the prevailing approach often entails using a majority voting mechanism. This mechanism includes assigning the class that receives the most number of votes across all trees to the given instance.

- **Out-of-Bag (OOB) Evaluation:**

The Out-of-Bag (OOB) evaluation method is used in academic research to assess the performance of machine learning algorithms. Given that each tree is trained on a distinct subset of the data, it follows that there are instances that were not used in the training process of a certain tree. The use of out-of-bag instances allows for internal validation, enabling an estimation of the model's performance without the need of an additional validation set.

- **Hyperparameter Tuning:**

The process of hyperparameter tuning involves optimizing the values of parameters that are not learned by the machine learning algorithm itself. Random Forests include hyperparameters that may be adjusted in order to enhance their performance. These hyperparameters include the number of trees, the maximum depth of each tree, and the minimum amount of samples necessary for a node split. The optimization of hyperparameters plays a pivotal role in attaining optimal classification outcomes.

The Random Forest classifier has several benefits within the domain of network intrusion detection.

- High Accuracy: Random Forests often exhibit exceptional precision when used for classification purposes. Machine learning algorithms are proficient in comprehending intricate connections among datasets, so rendering them very efficient in discerning between typical network activities and any unauthorized infiltrations.
- Resilience to Overfitting: The use of Random Forests, an ensemble learning technique that integrates numerous decision trees, serves to alleviate the issue of overfitting. The use of randomization techniques, such as feature selection and bagging (bootstrap aggregating), enhances the diversity and robustness of the models.
- Significance of Features: Random Forests provide a metric for determining the value of features by evaluating the frequency at which characteristics are used across the collection of trees. The provided information has significant value in discerning the most important elements that contribute to the identification of network intrusions.
- Managing Imbalanced Data: Network intrusion detection datasets often exhibit class imbalance, characterized by a substantial disparity between the number of intrusion occurrences and normal instances. Random Forests exhibit the capability to effectively manage unbalanced data by mitigating biases towards the majority class.
- Suitability for Large Datasets: Random Forests demonstrate effectiveness in effectively managing large and high-dimensional datasets, rendering them well-suited for network intrusion detection applications that include huge volumes of network data.
- Out-of-Bag Evaluation: Out-of-Bag Evaluation is a method used in machine learning to assess the performance of a model. The use of out-of-bag (OOB) data, which are excluded from the training process of each individual tree, allows for an internal estimation of the model's performance. This feature offers an integrated validation method, eliminating the need for a distinct validation set.
- Parallelization: The process of constructing individual decision trees inside a Random Forest may be parallelized, resulting in expedited training durations, particularly when handling a substantial quantity of trees.
- Versatility: Random Forests exhibit versatility as they may be effectively used for both classification and regression problems. In the domain of network intrusion detection, these systems possess the capability to be customized in order to identify and discern several categories of intrusions and abnormalities.
- Robustness against Noisy Data: Random Forests exhibit a higher level of resistance to the influence of noisy data points or outliers. The collective character of the concept contributes to noise reduction and enhances overall resilience.
- Absence of Requirement for Feature Scaling: Random Forests exhibit insensitivity towards the scale of input features, hence obviating the necessity for expensive preprocessing measures such as feature scaling. This streamlines the data preparation procedure.

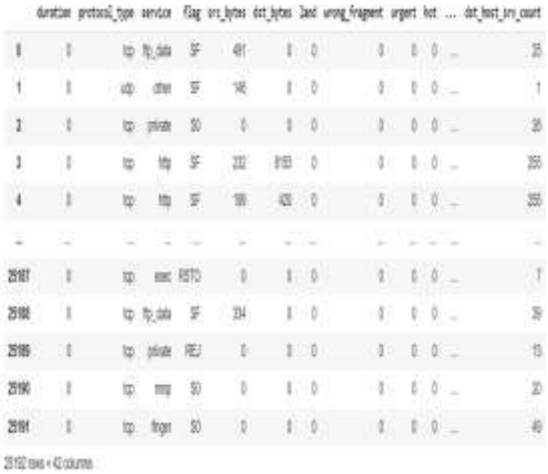
IV. EXPERIMENTAL RESULTS

This section presents an in-depth explanation of the outcomes derived from the simulations performed using the suggested technique. The dataset used in this investigation was obtained from Kaggle. The dataset was subjected to processing following the prescribed methodology. The dataset to be audited was given which comprises of a broad range of intrusions simulated in a military network environment. It developed an environment to obtain raw TCP/IP dump data for a network by imitating a typical US Air Force LAN. The LAN was concentrated like a genuine setting and bombarded with several assaults. A connection is a series of TCP packets that transport data to and from a source IP address to a destination IP address in accordance with a predetermined protocol. The packets start and stop at a certain time interval. Also, each connection is designated as either normal or as an attack with precisely one unique attack type. Each connection record consists of around 100 bytes.

For each TCP/IP connection, 41 quantitative and qualitative characteristics are retrieved from normal and attack data (3 qualitative and 38 quantitative aspects). The sample data from dataset is shown in figure 2.

The class variable has two categories:

- Normal
- Anomalous



	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent	hot	...	dst_host_srv_count
0	0	tcp	ftp_data	SF	491	0	0	0	0	0	...	25
1	0	udp	other	SF	146	0	0	0	0	0	...	1
2	0	tcp	private	S0	0	0	0	0	0	0	...	26
3	0	tcp	http	SF	232	8153	0	0	0	0	...	255
4	0	tcp	http	SF	199	420	0	0	0	0	...	255
...	...	...	...	...	...	...	...	...	...	...	...	...
25187	0	tcp	exec	RSTO	0	0	0	0	0	0	...	7
25188	0	tcp	ftp_data	SF	334	0	0	0	0	0	...	39
25189	0	tcp	private	REJ	0	0	0	0	0	0	...	13
25190	0	tcp	nnsp	S0	0	0	0	0	0	0	...	20
25191	0	tcp	finger	S0	0	0	0	0	0	0	...	49

25192 rows × 42 columns

Figure 2: Sample data from Dataset

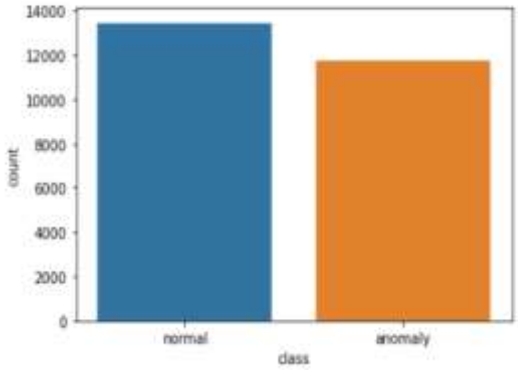
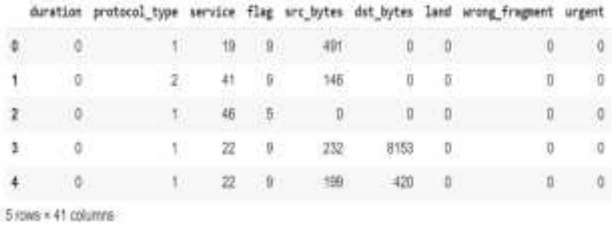


Figure 3: Count of classes in the train dataset

Figure 3 shows the count of the classes in the dataset. The count of classes denotes the quantity of examples or samples that are categorized under each unique class inside the dataset. The evaluation of dataset balance, or lack thereof, is a crucial component in comprehending its influence on the efficacy of machine learning models, particularly in tasks involving categorization.



	duration	protocol_type	service	flag	src_bytes	dst_bytes	land	wrong_fragment	urgent
0	0	1	19	9	491	0	0	0	0
1	0	2	41	9	146	0	0	0	0
2	0	1	46	5	0	0	0	0	0
3	0	1	22	9	232	8153	0	0	0
4	0	1	22	9	199	420	0	0	0

5 rows × 41 columns

Figure 4: Label Encoded dataset

Figure 4 shows the label encoded dataset. The use of Label Encoding is a methodology employed in the field of machine learning with the purpose of transforming categorical input into a numerical representation. In the aforementioned procedure, a distinct integer value is allocated to each distinct category or label. This conversion is especially advantageous when working with algorithms that need numerical input, since it empowers the model to correctly comprehend and handle categorical data.

Table 1: Comparative analysis

Methods	**Accuracy**
Logistic Regression Model	0.92
AdaBoost	0.97
Naive Bayes	0.89
SVM	0.97
Random Forest Classifier	**0.99**

Table 1 presents a comprehensive evaluation of several classification techniques, focusing on their respective levels of accuracy in relation to a certain job. The assessed techniques consist of a Logistic Regression Model achieving an accuracy of 0.92, AdaBoost achieving an accuracy of 0.97, Naive Bayes achieving an accuracy of 0.89, Support Vector Machine (SVM) achieving an accuracy of 0.97, and a Random Forest Classifier achieving the best accuracy of 0.99. The table presents the performance of each approach, with the Random Forest Classifier demonstrating the best accuracy compared to the other models examined.

V. CONCLUSION

In summary, this research has explored the significant domain of network intrusion detection, acknowledging the increasing complexity of cyber threats and the essential requirement for strong security measures. The utilization of Random Forest machine learning has been extensively investigated as a powerful option, owing to its proficiency in managing a wide range of intricate and varied network traffic patterns. The selected method exhibits a notable capacity to differentiate between regular and malicious behaviors, hence augmenting the effectiveness of network intrusion detection systems.

VI. REFERENCES

[1] Zaman, Marzia, and Chung-Horng Lung. "Evaluation of machine learning techniques for network intrusion detection." In *NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium*, pp. 1-5. IEEE, 2018.

[2] Verma, Abhishek, and Virender Ranga. "Statistical analysis of CIDDS-001 dataset for network intrusion detection systems using distance-based machine learning." *Procedia Computer Science* 125 (2018): 709-716.

[3] Sultana, Nasrin, Naveen Chilamkurti, Wei Peng, and Rabei Alhadad. "Survey on SDN based network intrusion detection system using machine learning

approaches." *Peer-to-Peer Networking and Applications* 12 (2019): 493-501.

[4] Liu, Hongyu, and Bo Lang. "Machine learning and deep learning methods for intrusion detection systems: A survey." *applied sciences* 9, no. 20 (2019): 4396.

[5] Jia, Yang, Meng Wang, and Yagang Wang. "Network intrusion detection algorithm based on deep neural network." *IET Information Security* 13, no. 1 (2019): 48-53.

[6] Aziz, Amira Sayed A., E. L. Sanaa, and Aboul Ella Hassanien. "Comparison of classification techniques applied for network intrusion detection and classification." *Journal of Applied Logic* 24 (2017): 109-118.

[7] Da Costa, Kelton AP, João P. Papa, Celso O. Lisboa, Roberto Munoz, and Victor Hugo C. de Albuquerque. "Internet of Things: A survey on machine learning-based intrusion detection approaches." *Computer Networks* 151 (2019): 147-157.

[8] Farnaaz, Nabila, and M. A. Jabbar. "Random forest modeling for network intrusion detection system." *Procedia Computer Science* 89 (2016): 213-217.

[9] Taher, Kazi Abu, Billal Mohammed Yasin Jisan, and Md Mahbubur Rahman. "Network intrusion detection using supervised machine learning technique with feature selection." In *2019 International conference on robotics, electrical and signal processing techniques (ICREST)*, pp. 643-646. IEEE, 2019.

[10] Abdulhammed, Razan, Hassan Musafer, Ali Alessa, Miad Faezipour, and Abdelshakour Abuzneid. "Features dimensionality reduction approaches for machine learning based network intrusion detection." *Electronics* 8, no. 3 (2019): 322.

[11] Mishra, Preeti, Vijay Varadharajan, Uday Tupakula, and Emmanuel S. Pilli. "A detailed investigation and analysis of using machine learning techniques for intrusion detection." *IEEE communications surveys & tutorials* 21, no. 1 (2018): 686-728.

[12] KarsligEl, M. Elif, A. Gökhan Yavuz, M. Amaç Güvensan, Khadija Hanifi, and Hasan Bank. "Network intrusion detection using machine learning anomaly detection algorithms." In *2017 25th Signal Processing and Communications Applications Conference (SIU)*, pp. 1-4. IEEE, 2017.

[13] Almseidin, Mohammad, Maen Alzubi, Szilveszter Kovacs, and Mouhammd Alkasassbeh. "Evaluation of machine learning algorithms for intrusion detection system." In *2017 IEEE 15th international symposium on intelligent systems and informatics (SISY)*, pp. 000277-000282. IEEE, 2017.

[14] Lee, Chie-Hong, Yann-Yean Su, Yu-Chun Lin, and Shie-Jue Lee. "Machine learning based network intrusion detection." In *2017 2nd IEEE International conference on computational intelligence and applications (ICCIA)*, pp. 79-83. IEEE, 2017.

[15] Belavagi, Manjula C., and Balachandra Muniyal. "Performance evaluation of supervised machine learning algorithms for intrusion detection." *Procedia Computer Science* 89 (2016): 117-123.