INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE

ISSN: 2057-5688

A Novel Framework for Detecting Network Intrusions Based on Machine Learning Algorithms

Gujju Bhaskar Rao, Research Scholar; Dr. Suribabu Potnuri, Professor Department of Computer Science and Engineering, J.S. University, Shikohabad, U.P. India and Dr. B. Laxmikantha, Associate Professor, Department of Computer Science and Engineering, Malla Reddy Institute of Engineering and Technology, Hyderabad, Telangana, India email: bhaskarrao.g@gmail.com

ABSTRACT

The efficiency of traditional rule-based network intrusion detection systems has grown increasingly questionable in the context of the ever-changing landscape of cyber threats. The findings of this study propose a novel framework for detecting network intrusions via the use of complex machine learning techniques. A dynamic, responsive, and intuitive approach to network security is prioritized by the method that has been offered, which is a departure from the rigidity of previous solutions. Several different types of machine learning models, including Convolutional Neural Networks (CNNs), Support Vector Machines (SVMs), and Random Forests, are used in this study. The purpose of this work is to examine a unique way for extracting and choosing characteristics. This strategy enables the model to focus on key potential threats while simultaneously lowering interference and improving the accuracy of detection. The performance of the framework has been evaluated in great detail via a series of experiments that were carried out using benchmark datasets. In many cases, the findings are superior to those obtained by traditional methods, demonstrating a significant enhancement in detection rates and a significant reduction in the number of false positives. Furthermore, the model that was powered by machine learning demonstrated its ability to adapt to new threat contexts, so confirming that it is suitable for use in real-world scenarios. The purpose of this research is to improve cybersecurity by offering new options for flexibility and resilience. This enhancement is accomplished by combining machine learning with network intrusion

ISSN: 2057-5688

detection. The framework offers a cutting-edge solution that is able to identify, collect information about, and adapt to emerging network incursions. As a result, it has the potential to influence the future of cyber security technologies.

Keywords—Attack detection; intrusion detection; machine learning; information security; artificial intelligence

1. INTRODUCTION

There has been an increase in the need for robust cybersecurity solutions as a result of the widespread dissemination of network systems and the growing dependence of many enterprises on these platforms. The detection of network intrusions in a timely manner is a crucial component of these methods. Throughout the course of history, the detection of these assaults has been accomplished via the use of rule-based procedures. Although these methods have been effective in some circumstances, they have been judged insufficient when faced with complex and ever-evolving cyber threats. When it comes to the field of cybersecurity, intrusion detection systems (also known as IDS) have a very important place. Intrusion Detection Systems (IDS) perform the role of attentive security guards by continuously monitoring network traffic and identifying any potential threats in a timely manner [3]. Signature-based and anomaly-based intrusion detection systems (IDS) are the two types of IDS that are used the most often. Intrusion Detection Systems (IDS) that are based on signatures are able to recognize familiar threats by comparing them to a database of threat signatures that has already been established. Intrusion detection systems (IDS) that are based on anomalies are able to identify potential dangers by recognizing deviations from the regular activity associated with a network. In spite of the fact that traditional methods provide some degree of security, the limits of these methods are becoming more apparent in the current cyber threat environment.

Due to the fact that they are dependent on the threat database that already exists, signature-based intrusion detection systems (IDS) are inherently limited in their capabilities. Their efficiency decreases when they are presented with new hazards that they are not acquainted with because of their poor capacity to recognize potential threats that are novel to them. On the other hand, anomaly-based intrusion detection systems (IDS), despite the

ISSN: 2057-5688

fact that they are theoretically capable of identifying new threats, often experience a high number of false positives. This is due to the fact that it is difficult to develop a clear definition of what constitutes 'normal' behavior.

In recent years, machine learning has garnered attention as a possible solution that may be used to alleviate these limits. The ability of machine learning to gain information from data and to make decisions based on that knowledge has shown significant promise in a number of different fields, including cybersecurity [9]. By using past data, continually improving their performance, and dynamically identifying new threats, machine learning technologies have the potential to particularly overcome the limitations that are associated with traditional intrusion detection systems (IDS). Through the use of the capabilities of machine learning, this study proposes a novel way to the identification of network intrusions. When it comes to network security, this architecture is superior than rule-based solutions since it offers a more flexible, adaptive, and instinctual approach.

This work was primarily motivated by the growing complexity of cyber threats and the consequent need for more advanced detection systems. This was the fundamental motivation for this investigation. Network incursions have evolved from relatively straightforward threats to complex attacks that are able to get beyond conventional security measures. This shift has occurred in the arena of network intrusions.

As a result, the primary objective of this research is to develop a system for detecting network intrusions that is driven by machine learning and is capable of properly recognizing and responding to both known and unknown threats. In comparison to traditional Intrusion Detection Systems (IDS), our objective is to make advantage of the predictive and adaptive capabilities of machine learning in order to achieve a higher level of accuracy in detecting and a lower rate of false-positive alerts. Through the implementation of this project, the issue of scalability in network intrusion detection would be addressed. The amount of network data that has to be monitored is increasing in tandem with the development and complexity of network systems. This presents a significant challenge for conventional Intrusion Detection Systems (IDS), which are designed to detect vulnerabilities in networks. Our technology, which makes use of machine learning, was developed with the express purpose of effectively managing the increased size and complexity that the situation brings about.

2. LITERATURE REVIEW

ISSN: 2057-5688

he literature review that is presented here provides a comprehensive overview of a number of machine learning strategies that are used in the area of network intrusion detection systems (IDS). This category of techniques include both the traditional algorithms and the emerging paradigms.

Decision Trees (DT) are often used in Intrusion Detection Systems (IDS) due to the fact that they are simple to comprehend and effective in the management of massive datasets. There has been a great performance shown by DT-based models, such as the C4.5 algorithm, in terms of both the accuracy of detection and the speed of detection. Nevertheless, they have a propensity to match the training data an excessive amount, which leads to insufficient generalization when they are presented with risks that are not before seen. Due to the fact that it is both basic and effective, the K-Nearest Neighbors (KNN) strategy is one of the most popular methods used in the area of Intrusion Detection Systems (IDS). The ability of this system to detect specific patterns is the key advantage it offers. As a result, it is particularly successful at identifying behavior that is not common within the population. However, when it is presented with high-dimensional data, which is a common occurrence in network intrusion detection, its effectiveness decreases.

Naive Bayes (NB) classifiers, which are derived from Bayes' theorem, are often used due to the fact that they are efficient in managing a significant number of characteristics. On the other hand, the idea of feature independence in Naive Bayes (NB) often leads to performance that is not sufficient. This is because the features in network traffic data are interrelated.

There is also an alternate approach known as Support Vector Machines (SVM), which is often praised for its remarkable accuracy and resistance to overfitting. On the other hand, support vector machines (SVMs) have two key drawbacks: they have a propensity to be sensitive to parameter choices and they have a high processing burden when working with large datasets.

Logistic regression, sometimes known as LR, is a statistical technique that is frequently used for binary classification problems, such as IDSS. The interpretability of logistic regression models is quite high, and they are also very good at dealing with noisy data. In spite of this, their performance is often below average when there are non-linear correlations present in the data.

Within the realm of intrusion detection systems (IDS), the Random Forest (RF) method is a popular example of an ensemble learning

ISSN: 2057-5688

approach. This approach integrates a large number of decision trees in order to reduce the risk of overfitting and improve the precision of predictions. A high-dimensional and large-scale data set may be successfully processed by Random Forest (RF), which is capable of doing so. On the other hand, if the characteristics included within the data have varied sizes, it may offer predictions that are inconsistent with reality.

An ensemble method known as AdaBoost, which is an abbreviation for Adaptive Boosting, is a method that combines weak classifiers in order to produce a strong classifier. The effectiveness of fundamental classifiers like Decision Trees (DT) and Naive Bayes (NB) has been improved with the use of AdaBoost in Intrusion Detection Systems (IDS). However, it is prone to data that is noisy and individuals who are outliers.The capacity of Artificial Neural Networks (ANN) to represent complex non-linear connections is a feature that is often seen in the operations of Intrusion Detection Systems (IDS). In order to gain information and make generalizations based on input data, Artificial Neural Networks (ANN) have been used in the building of Intrusion Detection Systems (IDS). One example of an ANN is the Multilayer Perceptron (MLP), which has been utilized in this process. On the other hand, they may need a significant amount of computer power and are frequently referred to as "black-box" models due to the fact that they are difficult to comprehend.

Within the realm of network intrusion detection, there has been a substantial rise in the number of research endeavors carried out over the course of the last few years. The purpose of these studies was to investigate a variety of machine learning algorithms, methods for picking features, and evaluation metrics in order to enhance the effectiveness of the intrusion detection system (IDS).

A novel framework that makes use of a broad variety of machine learning algorithms is presented in our research, in contrast to the studies that have been mentioned before. By using this strategy, it is able to efficiently embrace the complexities of network intrusion detection and exploit the benefits of each technique, hence overcoming the constraints that are associated with each technique individually. In addition, our system is designed to be able to adapt to the ever-changing nature of cyber threats, which strengthens its resistance to disruption and ensures its continued viability as a solution over the long run.

3. PROBLEM STATEMENT

An illustration of the flowchart of the proposed system may be seen in Figure 1. Figure 1 is an

ISSN: 2057-5688

illustration of the suggested structure, which is comprised of four phases that are sequentially arranged. The following is a list of the phases that are involved in this process: The first step is called "Data Cleaning," and it involves removing or correcting data entries that are irrelevant or incorrect. The second step is called "Data Transformation," and it involves normalizing and restructuring the data that has been cleaned. The third step is called "Feature Engineering," and it involves extracting and selecting significant attributes. Finally, the fourth step is called "Classification using Machine Learning," and it involves training models on the refined data to detect network security breaches.

4. Data Cleaning

A key technique that is aimed to remove inconsistencies, mistakes, and redundancies from the data, Data Cleaning is the initial step of the system that has been recommended. In order to accomplish this work, you will need to detect and resolve a large number of data anomalies, which may include data that is missing or incomplete, duplicate entries, inconsistent data formats, or incorrect data entries. Additionally, the process entails locating and treating outliers, since these anomalous data points have the potential to dramatically impact the subsequent analysis if they are not handled.

First and foremost, the purpose of this platform is to enhance the quality of the data, which will ultimately lead to an improvement in the efficiency and reliability of following operations. This phase is highly significant since the accuracy and quality of the data are critical aspects that influence the efficiency of any data-driven system hence it is very necessary that this step be taken. It ensures that the succeeding operations are not affected by any interference or inaccuracies in the data by establishing a solid foundation for them to be built upon. The end result of this stage is a dataset that has been improved and made more trustworthy. This dataset will serve as a basis for the succeeding phases of the system, which will be more exact and dependable.

5. Data Transformation

Immediately after the completion of the process of cleaning the data, the system proceeds to the next stage, which is the transformation of the data. It is at this phase that the data that has been processed is converted into a format that makes the subsequent stages of the system easier to understand. The normalization and restructuring processes are the two basic operations that are often included in this.

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688

The process of normalization is a technique that is used to equalize the scales of numerical characteristics inside a dataset. This ensures that all features have an equal effect on the analysis, regardless of the sizes that they were initially designed to have. When it comes to machine learning algorithms, it is very necessary for those that are influenced by the size of the input characteristics.

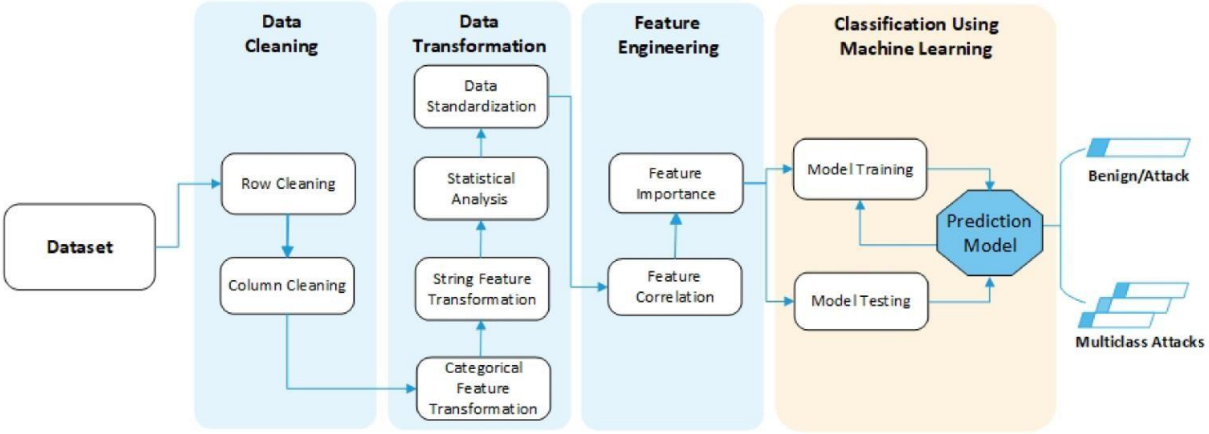


Fig. 1. Architecture of the proposed system for network intrusion detection

The process of changing the dataset into a format that is more suitable for feature engineering and machine learning is referred to as restructuring. Among the activities that may fall under this category are the conversion of category variables into numerical representations and the simplification of complex data structures in order to provide greater ease of processing by the system.

6. Feature Engineering

The phase of the system that is considered to be the most essential is the Feature Engineering phase. At this stage, the data that has been evaluated is assessed in order to identify the most significant qualities that must be included into the machine learning model as input. A comprehensive analysis of the data, the use of statistical methods, and the incorporation of specialized knowledge in a particular subject are all components of this process.

The transformation of the initial collection of characteristics into a new set of derived features that are predicted to provide a more accurate description of the underlying problem is accomplished via the process of feature extraction. It is possible that these additional

ISSN: 2057-5688

functions are less difficult, that they have more intricate linkages, or that they are more appropriate to the problem that is now faced.

The process of selecting the features that are most relevant from a set of original or extracted characteristics is referred to as feature selection. Techniques such as correlation analysis, mutual information, and wrapper methods are often used in the process of carrying out these duties. This strategy increases the computational efficiency and maybe boosts the performance of the machine learning model by lowering the dimensionality of the problem by the elimination of information that is either unneeded or duplicated.

7. Classification using Machine Learning

The last step of the system is called Classification using Machine Learning, and it makes use of the results of the phases that came before it in order to identify patterns in the data and categorize network activity as either normal or invasive. This step involves the use of a specific machine learning algorithm, which is then trained using the data that has been processed and obtained from the stages that came before it.

Entering the feature vectors into the model is part of the training phase. This gives the model the opportunity to learn about the connections that exist between the features and the variable that is being trained on. After the training phase of the model has been finished, it may be assessed for its performance by being tested with new data that it has not seen previously.

The primary objective of this stage is to achieve the development of a prediction model that is capable of reliably and efficiently detecting network intrusions. It is important to note that the proposed system is heavily reliant on this model as its fundamental component, and the effectiveness of the model has a direct impact on the overall performance of the system. This step is comprised of many essential components, including the selection of a machine learning algorithm, the adjustment of its parameters, and the evaluation of its performance.

8. DATASET

Within the realm of network intrusion detection research, the NSL-KDD dataset is a well-known benchmark dataset that is used. This is an enhanced version of the KDD'99 dataset, which is widely acknowledged as being of great significance in the area. MIT Lincoln Labs was responsible for carrying out the DARPA Intrusion Detection Evaluation

ISSN: 2057-5688

Program in 1998, which is where the KDD'99 dataset was created from. In addition to providing a realistic depiction of network traffic statistics, it also simulates a wide variety of attacks.

.

9. Description of the Dataset

It was acknowledged that the KDD'99 dataset had a number of significant issues, despite the fact that it was very helpful to academics. The inclusion of a significant number of duplicate entries, which led to an artificial skew in the system, was one of these issues. Another problem was the distribution of the different types of network invasions, which was not feasible. As an improved alternative, the NSL-KDD dataset was presented to researchers as a means of overcoming these limitations and providing them with a more accurate environment to work in.
A more realistic and balanced depiction of network traffic is achieved as a consequence of the NSL-KDD dataset, which improves upon the original dataset by removing things that are duplicated at the same time. A total of around 125,973 records are included in the dataset for training purposes (KDDTrain+) and 22,544 records are included for testing purposes (KDDTest+), which includes a variety of invasions. A visual representation of the NSL-KDD dataset is shown in Figure 2, which illustrates the distribution of the many classes that are included in the dataset. A total of around 53% of the instances in the dataset are classified as "normal," which indicates that they are authorized network actions. The other 47% of the occurrences are classified as various types of "attack" classifications. Furthermore, Figure 2 presents a detailed study of the NSL-KDD dataset, with a particular emphasis on the breakdown of methods included in the dataset. The distribution is made up of data from the TCP protocol, which accounts for 82% of the total, followed by data from the UDP protocol, which accounts for 12%. The remaining seven percent is accounted for by data from the ICMP protocol.
A variety of different kinds of network attacks are denoted by each category. The purpose of denial of service attacks is to make a computer or network resource unavailable when they are launched. To get illegal root access, U2R attacks make advantage of vulnerabilities in the system.

ISSN: 2057-5688

When attempting to get local access, R2L attacks take use of flaws. One method of conducting probe attacks is to scan a network in order to gather information or locate vulnerabilities that are already known.

In Figure 3, a visual depiction of the distribution of 'flags' in the NSL-KDD dataset is shown. More particularly, a comparison is made between the 'normal' and 'attack' classes. During the process of network communication, flags are used to indicate the current status of a particular connection, as well as to symbolize various kinds of events or failures. The distribution of these flags across a variety of classes is of utmost significance because they have the potential to serve as powerful indicators of anomalous or malicious behavior in the data that is captured by the network.

This graphic is an illustration of a comparative study that visually depicts the distribution of unique flag values between instances that are labeled as "normal" and occurrences that are classified as "attack." The presentation of the data in this manner makes it possible to have a more in-depth understanding of the correlation that exists between the flag values and the class of the link. Therefore, this may give significant views on the possible correlations between different flag values and diverse network traffic patterns, as well as their potential uses in spotting network intrusions. Moreover, this may also provide new insights into the prospective applications of these correlations.

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688

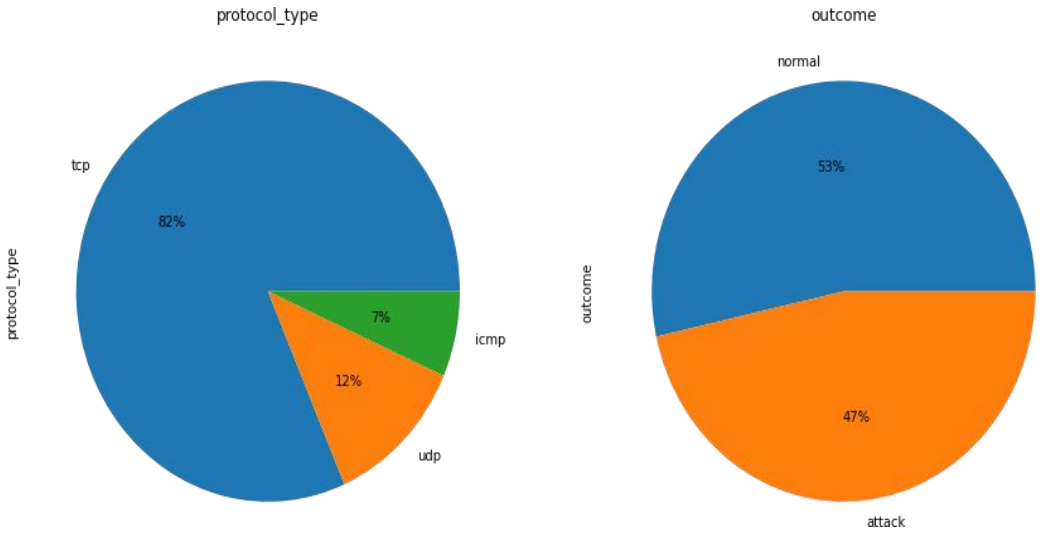


Fig. 2. General description of the NSL-KDD dataset

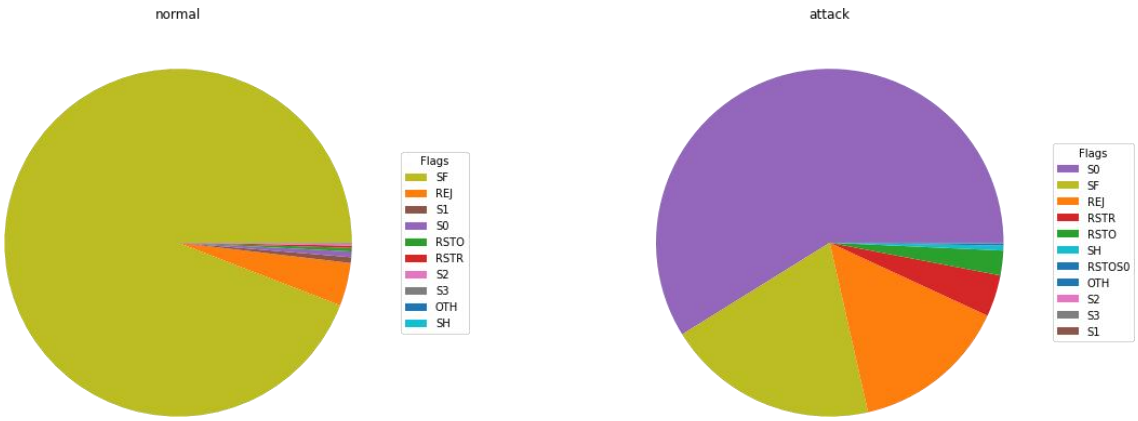


Fig. 3. Flags of normal and attack classes in the dataset

The efficiency of employing flags as indications of malicious behavior may vary depending on a number of variables, such as the characteristics of the network environment and the sorts of assaults that are most prevalent. The distribution that has been shown may be used as a foundation for

ISSN: 2057-5688

further study and debate on the significance of flags in the domain of network intrusion detection.

There is a binary class label that represents the target variable, and it indicates whether the connection was categorized as a regular connection or an assault. In addition to this, it includes a detailed description that provides a specific indication of the kind of attack that was committed, especially if it included unlawful entry. Due to the fact that it contains a wide variety of attack types and a thorough collection of characteristics, the NSL-KDD dataset is very useful for the construction and evaluation of network intrusion detection systems.

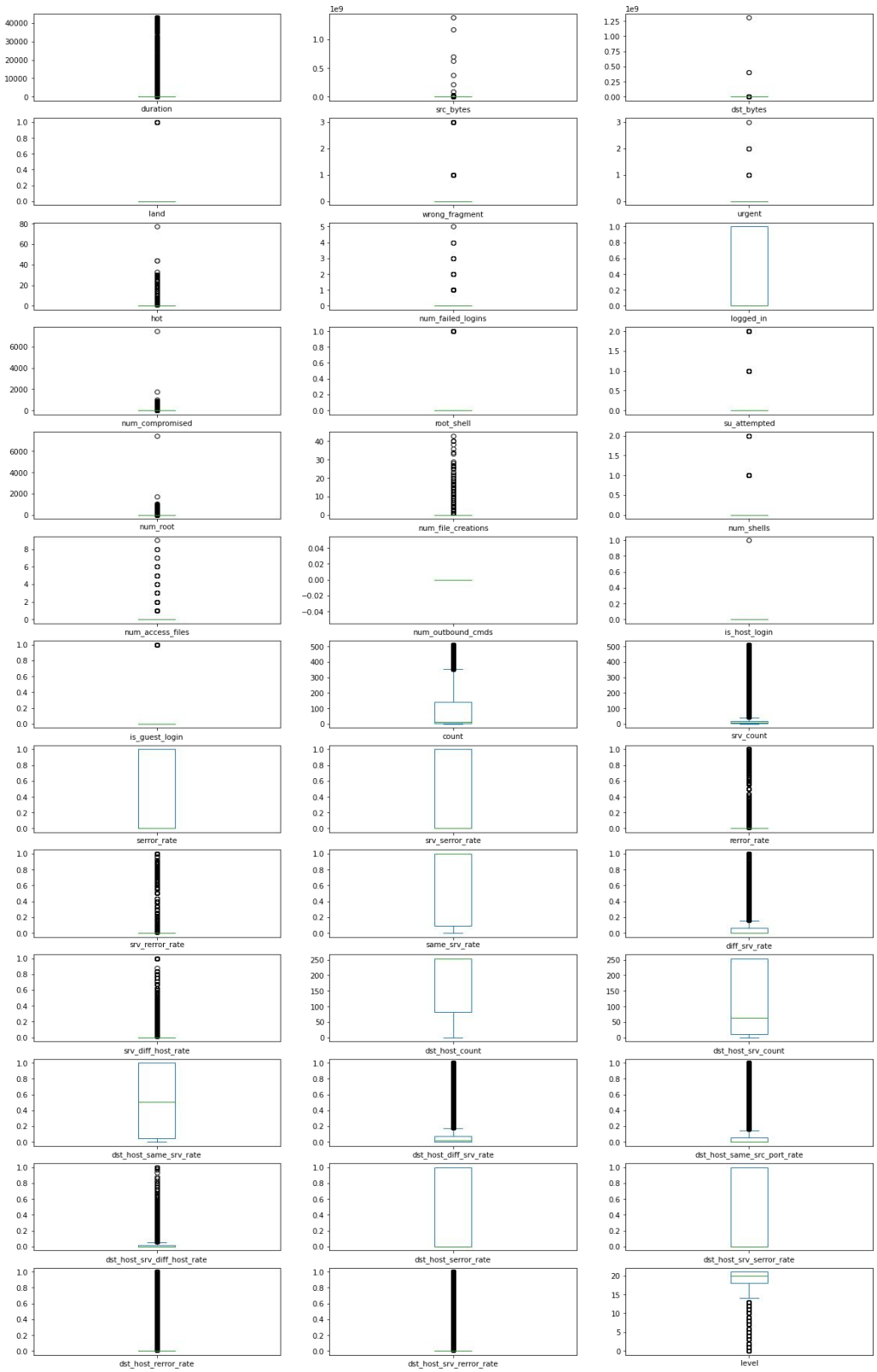
An realistic and rigorous environment for testing machine learning algorithms is provided by the NSL-KDD dataset. This environment makes it possible to conduct an exhaustive examination of the efficacy of these algorithms in performing intrusion detection tasks. In spite of the significant progress that has been made in this area, the NSL-KDD dataset continues to be an important and suitable option for academics. This is because it continues to provide essential insights on the effectiveness of a variety of approaches and systems. That being the case, it serves as an excellent resource for our investigation, providing a comprehensive and varied dataset that can be used to assess the efficiency of the strategy that we have presented.

10. Data Preprocessing

In order to evaluate the distribution of the data and locate outliers in each column of the NSL-KDD dataset, we used the boxplot method, which is seen in Figure 4. Through the use of visual representations, we are able to easily recognize the interquartile ranges, recognize any potential outliers, and get a full grasp of the overall distribution of the data. Figure 3 is a useful visual tool that assists us in learning the statistical complexities of the dataset that we have chosen. This enables us to effectively discover any data points that indicate abnormality.

.

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688



ISSN: 2057-5688

Fig. 4. Box plots for each feature

11. EXPERIMENTAL RESULTS

The next section, which is headed "Experiment Results," offers a detailed summary of the data that were obtained from our research study. The purpose of this investigation was to evaluate the effectiveness of the novel approach that was offered for spotting network intrusions. Several different machine learning algorithms were used and tested for the goal of network intrusion detection. These techniques were utilized by making use of the NSL-KDD dataset. In order to provide a comprehensive analysis of the results, the performance of the models was assessed by using significant metrics like as accuracy, precision, recall, and F-score. The purpose of this section is to more clearly

$$accuracy = \frac{TP + TN}{TP + FN + TN + FP} \quad (1)$$

In this research, experimental data are presented that shed light on the relative usefulness of the approaches that were being used. In addition, it highlights the potential advantages and limitations of the framework that was provided when applied to a real-world scenario.

When it comes to classification tasks, precision is an essential parameter that analyzes the model's capacity to produce accurate predictions of positive instances. Precision may be defined as the ratio of the number of genuine positives to the sum of the number of false positives and truthful positives. As the accuracy of the model increases, it suggests that the positive predictions it makes are very precise, hence minimizing the likelihood of producing false positives. Text provided by the user is referred to as "context."

There are six distinct machine learning models that are used in the demanding subject of network intrusion detection, and Figure 5 depicts confusion matrices for each of these models. An essential visual tool that provides a condensed summary of the performance of a confusion matrix is each confusion matrix.

$$precision = \frac{TP}{TP + FP} \quad (2)$$

In the realm of classification problems, recall, which is sometimes referred to as

ISSN: 2057-5688

sensitivity or true positive rate, is an important component of the measurement process [32]. It is the capacity of the model to reliably identify all relevant events that is being measured by the metric. The calculation is carried out by dividing the total number of true positives by the sum of all the true positives and false negatives involved. When the recall is higher, it suggests that there are fewer false negatives, which ensures that the model is able to capture the bulk of the positive data.

$$recall = \frac{TP}{TP + FN} \quad (3)$$

When evaluating the performance of the model, it is not sufficient to simply provide the number of correct and wrong predictions generated by the model. In addition to this, it places an emphasis on the particular categories of errors that were made. By making use of these vast insights, we are able to carefully analyze the efficacy of each model in recognizing network intrusions, which enables us to facilitate a data-driven selection of the methodology that is the most effective.

The Receiver Operating Characteristic (ROC) curves and the Area Under the Curve (AUC) values that correlate to them are shown in Figure 6. These curves are for six different machine learning models that were used in the Network intrusion detection application.

There is a complete statistic known as the F-score, which is also widely referred to as the F1 score [33]. This statistic combines accuracy and recall into a single measurement. In order to get the harmonic mean of accuracy and recall, both metrics are given equal weight in the calculation. When the F-score is high, it implies that the model has reached a high level of accuracy and recall, which shows that there is a perfect balance between the number of false positives and false negatives.

$$F1 = \frac{2 \cdot precision \cdot recall}{precision + recall} \quad (4)$$

A receiver operating characteristic (ROC) curve is a graphical representation that illustrates the connection between the true positive rate (sensitivity) and the false positive rate (1-specificity) for various threshold values. Specifically, it offers a crystal clear demonstration of the trade-off that exists between sensitivity and

ISSN: 2057-5688

specificity for the models that are being studied. The area under the curve (AUC), on the other hand, provides a thorough assessment of the performance of the model across all possible classification considerations.

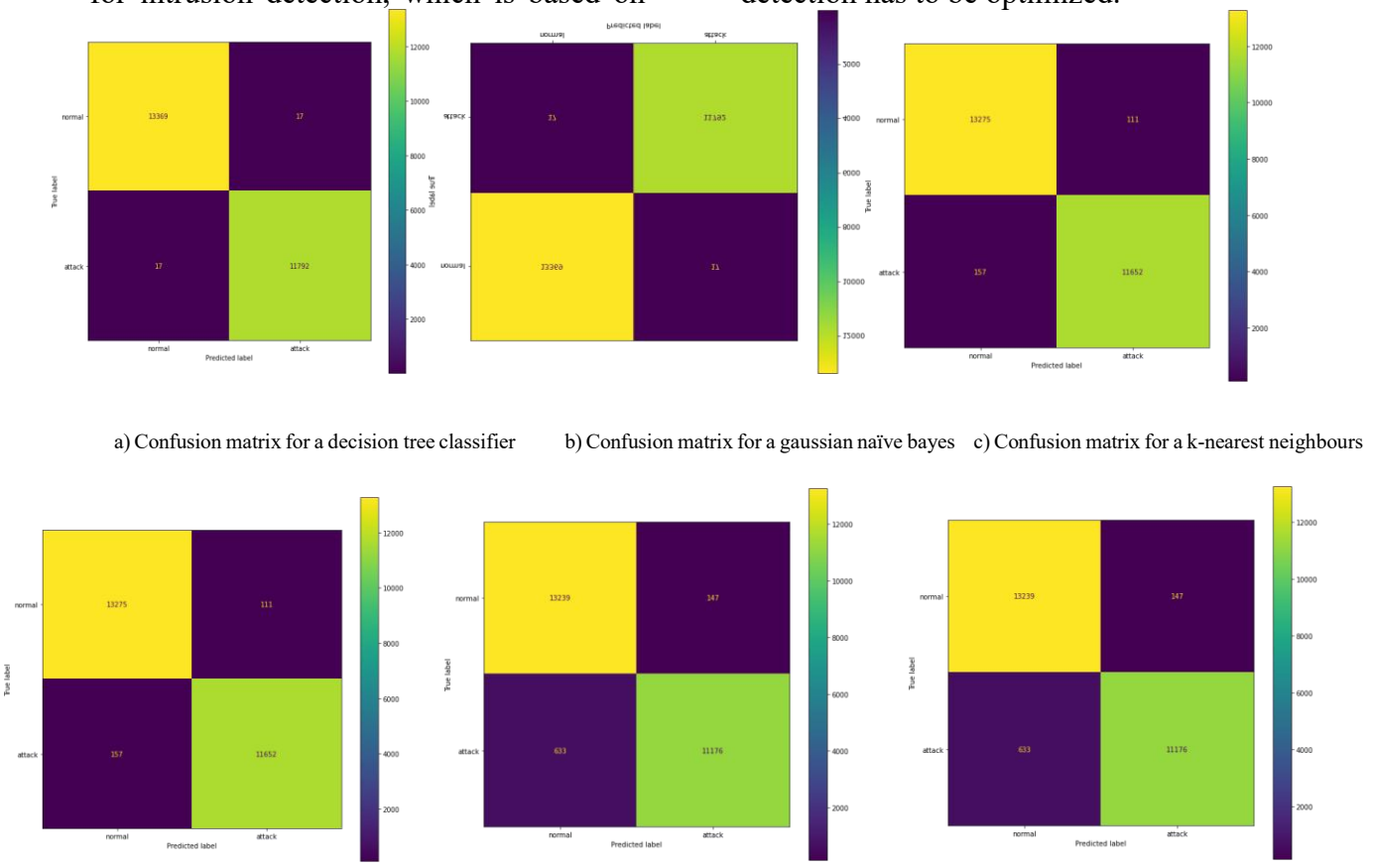
Through the examination of the ROC-AUC plots, we are able to conduct a comparative evaluation of the performance of the models, which results in the provision of insights into the relative effectiveness of the models in distinguishing between normal and attack events in network traffic data.

For a comprehensive evaluation of the performance of a classification model, it is essential to make use of the evaluation metrics that have been mentioned above. These evaluation measures include accuracy, precision, recall, and F-score. Accuracy is a measurement of the proportion of right predictions that are produced by the model, while precision evaluates the model's ability to reliably identify positive instances. The capacity of the model to recognize all important events is evaluated using the recall metric, whilst the F-score serves as an all-encompassing statistic that takes into consideration both the accuracy and the recall metrics. These data, when taken as a whole, provide a full assessment of a model's capacity to produce correct predictions. The relevance of each individual measurement varies according to the specific objective of the classification assignment.

The determination of the most promising model for intrusion detection is made easier as a result of this extensive study. The findings of the experiments indicate that the k-Nearest Neighbors (kNN) technique is better in terms of its effectiveness in detecting network intrusions. This is shown by the fact that it has superior metrics in the assessment parameters categories. On the other hand, it is possible to see some indications of overfitting in the performance of the Random Forest classifier as well as the Support Vector Machines (SVM) classifier. One of the mistakes that may occur in modeling is called overfitting. This occurs when a function is overly aligned with a limited collection of data points, which hinders the ability of the model to apply to new data. It is possible that the propensity

INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688

of these classifiers to overfit might reduce their effectiveness in a network intrusion detection scenario that takes place in the real world. The whole ROC-AUC curves of the six different techniques that were applied show that the proposed framework for intrusion detection, which is based on machine learning algorithms, is applicable and can be put into practice. It is possible that the most optimum model with a ROC-AUC curve will be the principal algorithm in an intrusion detection system. This is because the framework for intrusion detection has to be optimized.

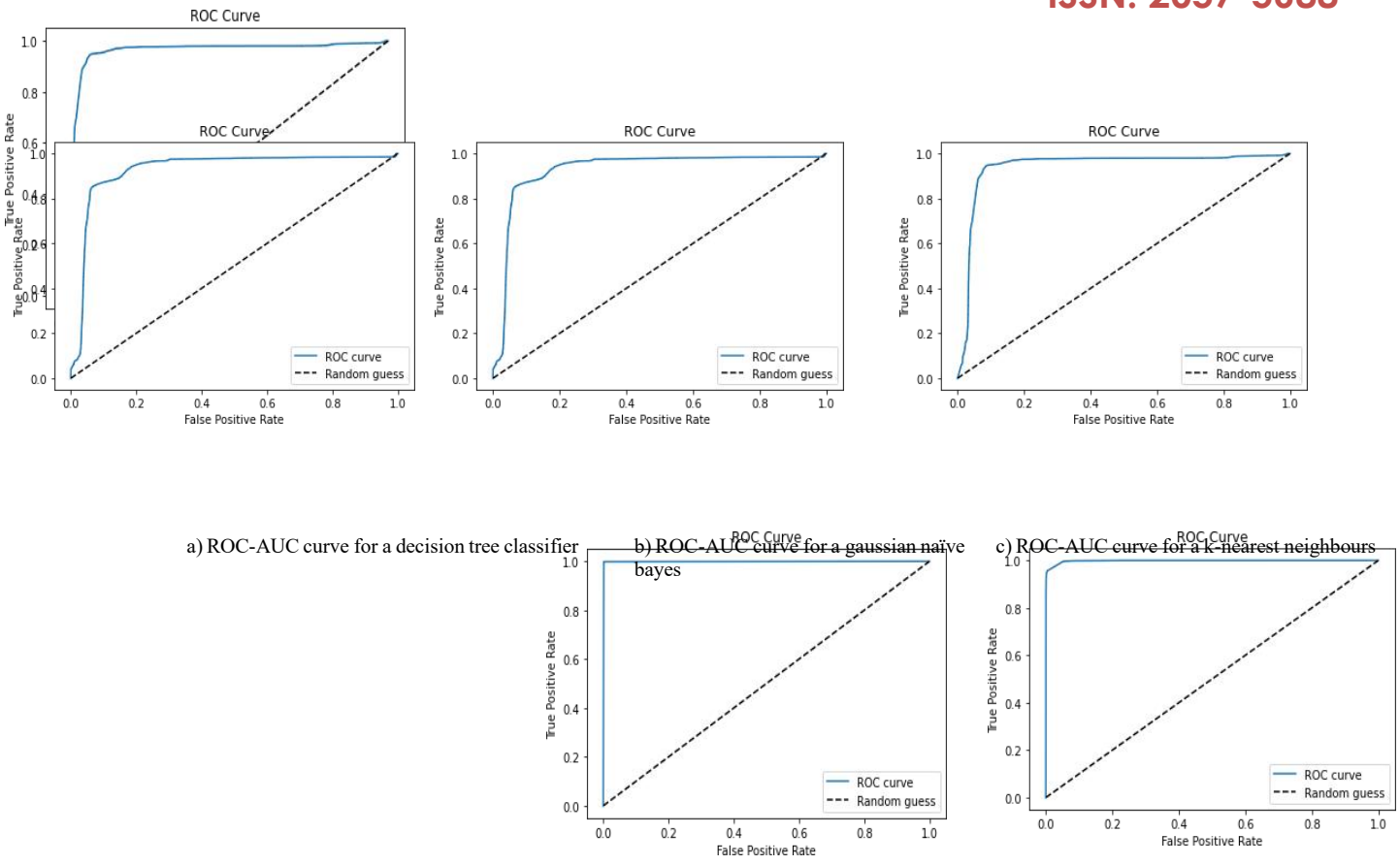


a) Confusion matrix for a decision tree classifier b) Confusion matrix for a gaussian naïve bayes c) Confusion matrix for a k-nearest neighbours

d) Confusion matrix for a logistic regression e) Confusion matrix for a random forest classifier f) Confusion matrix for a SVM

Fig. 5. **Confusion matrices for machine learning methods in network intrusion detection problem**

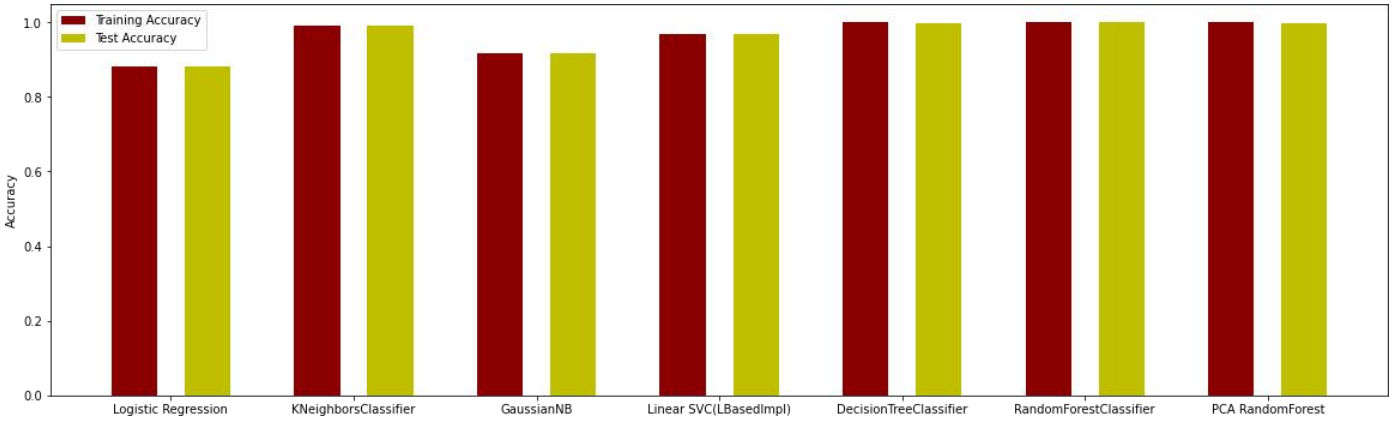
INTERNATIONAL JOURNAL OF TECHNO-ENGINEERING
IJTE
ISSN: 2057-5688



a) ROC-AUC curve for a decision tree classifier b) ROC-AUC curve for a gaussian naïve bayes c) ROC-AUC curve for a k-nearest neighbours

d) ROC-AUC curve for a logistic regression e) ROC-AUC curve for a random forest classifier f) ROC-AUC curve for a support vector machin

Fig. 6. AUC-ROC curves for machine learning methods in network intrusion detection problem



ISSN: 2057-5688

Fig. 7. Train and test accuracies of machine learning methods in network intrusion detection problem

Figure 7 shows a comparative depiction of the training and test accuracies derived from the machine learning algorithms used for network intrusion detection on the NSL-KDD dataset. These approaches were employed to discover vulnerabilities in the network.

One of the metrics that is used to measure how well a model works on the training dataset is called training accuracy. This statistic indicates that the model is able to properly reflect the data that is presented. The test accuracy, on the other hand, analyzes the performance of the model on a dataset that it has not before seen. This demonstrates that the model is able to apply its knowledge to new data that is beyond what it was trained on. The graph that is shown in Figure 7 gives us the opportunity to investigate the connection that exists between these two categories of accuracy for each method that is used. Through the process of contrasting the accuracy of the training set with that of the test set, we may be able to get useful information on the possibility of overfitting or underfitting. Understanding the effectiveness of the models requires that you have these information. Overfitting is often indicated when the accuracy of the training performance is high while the accuracy of the test performance is low. A poor training accuracy, on the other hand, may be an indication of underfitting conditions.

The purpose of this comparative assessment is to offer information that can be used for future decisions about the selection of models, as well as to examine whether or not there is a potential need for modifications to the complexity of the models or training methods in order to improve performance.

Using the NSL-KDD dataset, Figure 7 provides a comparison of the training and test accuracies that were attained by the machine learning approaches that were used for the purpose of intrusion detection.

ISSN: 2057-5688

In training precision, the ability of the model to accurately predict positive occurrences within the training dataset is measured, while in test precision, the performance of the model is evaluated on data that it has not before seen. In the event that we examine the training and test precisions that are shown in Figure 8, we could be able to get significant insights about the possibility of overfitting or underfitting occurring

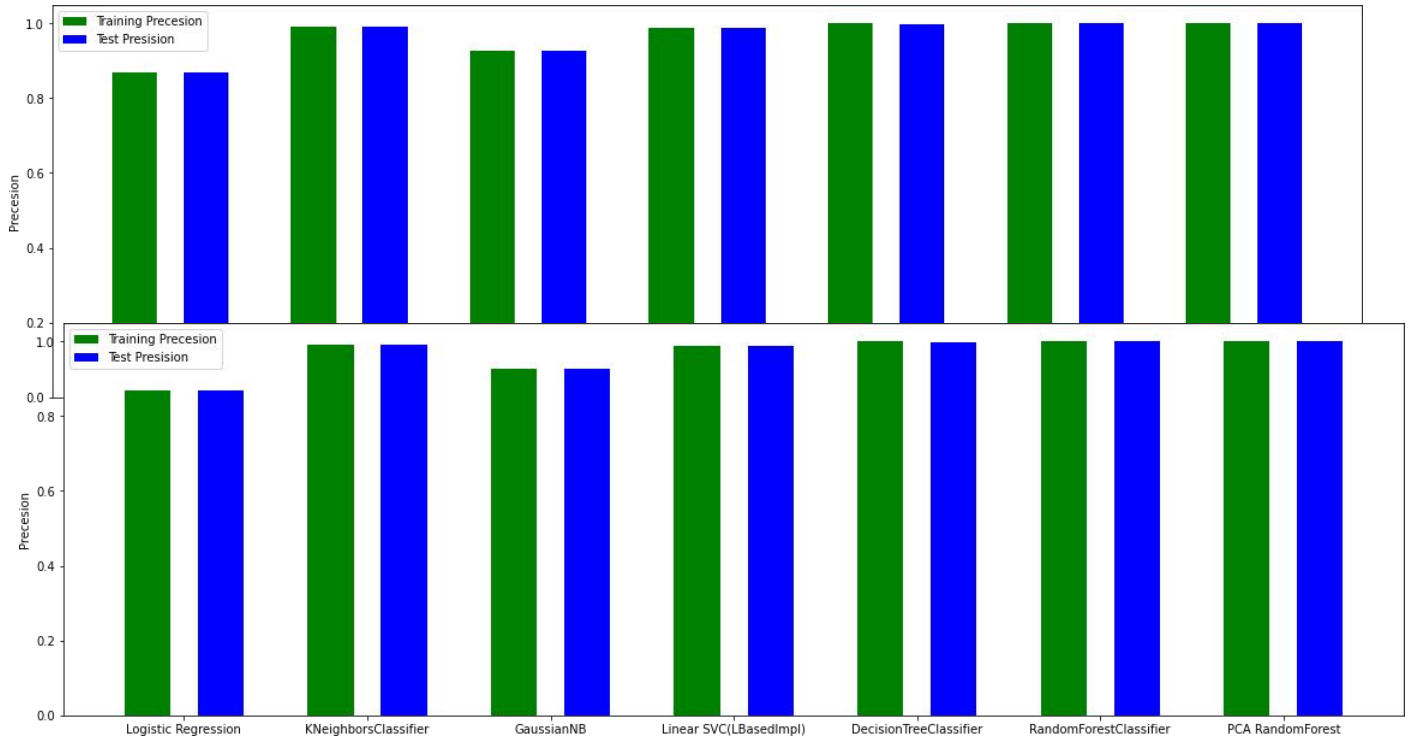


Fig. 8. Train and test precisions of machine learning methods in network intrusion detection problem

Fig. 9. Train and test recall of machine learning methods in network intrusion detection problem

ISSN: 2057-5688

The ability of the models to generalize and properly identify network intrusions may be evaluated by conducting an analysis of the connection between the correctness of training data and test data during the training phase. We have a better understanding of the performance of the model thanks to the data that is shown in Figure 8. These data also give direction for picking the most suited techniques for intrusion detection tasks. Figure 9 provides a comprehensive comparison of the training and test recalls that were acquired from the machine learning approaches that were used in area of intrusion detection by making use of the NSL-KDD dataset

Training recall is an evaluation of the model's ability to properly identify positive occurrences in the training dataset, while test recall is an evaluation of the model's performance on fresh data that is unknown to the model. We may be able to get a more in-depth knowledge of the models' capacity to generalize and successfully detect significant instances of network intrusions by conducting an analysis of the link between training and test recalls, which is shown in Figure 9.

It is possible that the existence of overfitting or underfitting is indicated by differences in recall performance between the data used for training and those used for testing. When there is a high recall rate during training but a significant fall in recall during testing, this may be an indication that the model is being overfit. In this particular scenario, it is essential to execute actions that would enhance generality. If, on the other hand, the training and test recalls are low, this may be an indication of underfitting. This would imply that the model needs to be improved or that the feature selection needs to be reevaluated.

The findings that are shown in Figure 9 improve one's understanding of the performance of the model and provide assistance in selecting appropriate methods for effective intrusion detection in real-world scenarios by making use of the NSL-KDD dataset.

12. DISCUSSION

ISSN: 2057-5688

Important insights into the efficacy and comparative performance of the proposed new framework for detecting network intrusions using machine learning techniques have been gleaned from the data that were obtained from the tests that were carried out on the framework. In this discussion, we will examine the most important findings, highlight the benefits and limitations of the framework, and suggest potential avenues for further improvement in the future.

Based on the empirical results, it was determined that the k-Nearest Neighbors (kNN) approach shown remarkable effectiveness in detecting network intrusions on the NSL-KDD dataset. This is evident from the fact that its accuracy, precision, recall, and F-score values are much higher than those of the other procedures that were used. It is possible that the success of the kNN approach in this particular setting might be related to its capacity to identify similar occurrences by taking into account their proximity in the feature space. This is accomplished without the need to depend on specific assumptions about the distribution of the data that is being used. In the process of handling network intrusion detection assignments, the results highlight the capacity of instance-based approaches such as kNN.

On the other hand, the Random Forest classifier and the Support Vector Machines (SVM) showed indications of overfitting. Overfitting occurs when a model conforms to the training data in an excessive manner, which results in an insufficient capacity to predict outcomes for fresh data from which the model has no prior experience. The process of machine learning often encounters challenges, one of which is the management of complex datasets, such as network intrusion detection [34]. Increasing the ability of the models to generalize is one of the numerous ways that may be used in order to solve the issue of overfitting on the models. Regularization, feature selection, and hyperparameter tuning are some of the approaches that fall under this category.

In addition, the outcomes of the research offered useful insights into the effectiveness of several additional strategies that were put into practice. kNN displayed performance that was similar to that of Decision Tree, Gaussian Naive Bayes, and Logistic Regression; nevertheless, their performance was slightly lower than that of kNN. It is possible that the suitability of each of these solutions will vary depending on the specific requirements and characteristics of the intrusion detection problem that is being addressed. Each of these solutions has a unique set of benefits and

ISSN: 2057-5688

drawbacks. There is a possibility that more research into these methods, which may include ensemble procedures like as AdaBoost, might result in improved results.

The fact that the framework makes use of the NSL-KDD dataset, which is a benchmark dataset that is well known and respected, contributes to the results' reputation for reliability. It is possible to conduct a realistic evaluation of the suggested architecture thanks to the comprehensive collection of characteristics and the broad variety of network intrusion types that are included within the dataset. In spite of this, it is of the utmost importance to acknowledge that the NSL-KDD dataset does, in fact, have certain limitations, such as the use of preprocessed data and the likelihood of bias resulting from the processes of data collection [35]. It is recommended that future study take into consideration the incorporation of additional datasets and actual network traffic in order to better validate the usefulness of the framework in situations that are applicable to real-world situations.

The capacity of the models to generalize was evaluated by comparing their accuracies, precisions, recalls, and F-scores throughout training and during testing. This comparison yielded useful information. The existence of overfitting or underfitting may be indicated by differences in performance metrics between the data collected during training and those collected during testing [36]. The models that demonstrate great performance during training but much poorer performance on the test set are the ones that should be the primary focus of attention since this indicates that the models have been overfit. Methods like as regularization techniques, cross-validation, and early stopping are examples of tactics that have the potential to reduce the effects of overfitting and enhance the generalization of models.

To add insult to injury, the findings suggest that it is essential to choose and modify the hyperparameters for each machine learning algorithm with great care [37]. There is a possibility that the performance of the algorithms may be significantly impacted by modifying the parameters of the algorithms, such as the maximum depth of decision trees or the number of neighbors in k-nearest neighbors [38]. There is a possibility that more improvements in performance might be achieved by carrying out a comprehensive search for hyperparameters by using techniques such as grid search or Bayesian optimization [39].

It is essential to acknowledge the limitations of the proposed paradigm, despite the fact that it produced result that was positive. At first, the evaluation was performed on a specific dataset; however, it is possible that the efficacy will

ISSN: 2057-5688

change when it is applied to other datasets or to network configurations that are applicable in the real world. It is necessary to do more research in order to find out whether or not the framework is capable of adapting to different network architectures, traffic patterns, and attack scenarios [40]. Additionally, the framework focused primarily on supervised learning approaches [41], ignoring the possible benefits of unsupervised or semi-supervised methods in network intrusion detection [42]. This was done in order to accomplish the framework's primary objective. It is possible that further research may make use of hybrid models or anomaly detection approaches in order to further enhance the capabilities of the said system.

In a nutshell, the experimental results that were shown in this study highlight the effectiveness of the novel framework that was proposed for spotting network intrusions via the use of machine learning methods. When compared to other algorithms that are currently being used, the k-Nearest Neighbors method has shown greater performance in terms of accuracy, precision, recall, and F-score levels. In order to ensure the highest possible level of performance, the results highlight the necessity of paying close attention to the selection of models, making adjustments to hyperparameters, and minimizing overfitting issues. The findings contribute to the advancement of existing knowledge in the field of network intrusion detection and serve as a foundation for further research and the creation of robust and adaptable organizational structures for network security. Learning by machine has been used in a variety of fields, ranging from the medical field to the field of smart cities [43-45]. Techniques from the field of machine learning were used in this investigation to solve the problem of network intrusion detection. Based on the facts that were gathered, it is clear that machine learning is also quite effective in this particular field.

13. CONCLUSION

Within the scope of this study, a novel framework for spotting network intrusions via the use of machine learning methods is presented. The NSL-KDD dataset, which is well renowned, is used in order to evaluate the performance of the system. The results of the experiments demonstrated beyond a reasonable doubt that the framework is effective in identifying network intrusions. The k-Nearest Neighbors (kNN) algorithm emerged as the most successful solution to this problem. The comprehensive evaluation metrics of the framework, which included accuracy, precision, recall, and F-score, provided a comprehensive analysis of the framework's functional capabilities.

ISSN: 2057-5688

The results highlight the need of selecting appropriate machine learning algorithms with care and adjusting their hyperparameters in order to achieve the highest potential level of performance in network intrusion detection tasks. It is possible that the success of the kNN approach might be attributed to its ability to make advantage of proximity-based learning and efficiently manage nuanced patterns within the dataset. In addition, the comparison of the accuracies, precisions, recalls, and F-scores of the training and test versions brought to light the possibility of overfitting issues and highlighted the need of generalizing the model.

Despite the fact that the proposed paradigm produced potentially positive results, it is essential to acknowledge the constraints it has. The majority of the evaluation was performed on the NSL-KDD dataset; however, the effectiveness of the approach may be altered when it is applied to different datasets or to network configurations that are seen in the real world. Furthermore, the framework placed a primary emphasis on supervised learning methods, however, it failed to take into account the possible benefits that may be gained from unsupervised or semi-supervised approaches. There is a possibility that the framework's capability to recognize attacks that have not been seen previously might be enhanced by the investigation of hybrid models and the incorporation of anomaly detection technologies.

For the purpose of providing a realistic solution for detecting network intrusions via the use of machine learning methods, the creative framework that was provided gives a solution. The results of the experiments demonstrate that the strategy is effective and highlight the relevance of selecting algorithms with great care and optimizing hyperparameters. There is the possibility that the framework might serve as a foundation for doing more research in the development of intrusion detection systems that are both robust and adaptable. This would ensure the protection of network security in an environment that is always changing and containing possible threats. In further study, the evaluation should be expanded to include a greater number of datasets, and the use of unsupervised and semi-supervised approaches should be investigated in order to improve both performance and flexibility.

14. REFERENCES

1) Zhang, C., Chen, Y., Meng, Y., Ruan, F., Chen, R., Li, Y., & Yang, Y. (2021). A novel framework design of network intrusion detection based on machine

ISSN: 2057-5688

learning techniques. Security and Communication Networks, 2021, 1-15.

2) Ahmad, Z., Shahid Khan, A., Wai Shiang, C., Abdullah, J., & Ahmad, F. (2021). Network intrusion detection system: A systematic study of machine learning and deep learning approaches. Transactions on Emerging Telecommunications Technologies, 32(1), e4150.

3) Saheed, Y. K., Abiodun, A. I., Misra, S., Holone, M. K., & Colomo- Palacios, R. (2022). A machine learning-based intrusion detection for detecting internet of things network attacks. Alexandria Engineering Journal, 61(12), 9395-9409.

4) Musleh, D., Alotaibi, M., Alhaidari, F., Rahman, A., & Mohammad, R.

5) M. (2023). Intrusion Detection System Using Feature Extraction with Machine Learning Algorithms in IoT. Journal of Sensor and Actuator Networks, 12(2), 29.

6) Mighan, S. N., & Kahani, M. (2021). A novel scalable intrusion detection system based on deep learning. International Journal of Information Security, 20, 387-403.

7) Awad, N. A. (2021). Enhancing Network Intrusion Detection Model Using Machine Learning Algorithms. Computers, Materials & Continua, 67(1).

8) Sultanovich, O. B., Ergeshovich, S. E., Duisenbekovich, O. E., Balabekovna, K. B., Nagashbek, K. Z., & Nurlakovich, K. A. (2016). National Sports in the Sphere of Physical Culture as a Means of Forming Professional Competence of Future Coach Instructors. Indian Journal of Science and Technology, 9(5), 87605-87605.

9) Sivanantham, S., Mohanraj, V., Suresh, Y., & Senthilkumar, J. (2023). Association Rule Mining Frequent-Pattern-Based Intrusion Detection in Network. Computer Systems Science and Engineering, 44(2), 1617- 1631.

10) Apruzzese, G., Pajola, L., & Conti, M. (2022). The cross-evaluation of machine learning-based network intrusion detection systems. IEEE Transactions on Network and Service Management.

11) Alzahrani, A. O., & Alenazi, M. J. (2021). Designing a network intrusion detection system based on machine learning for software defined networks. Future Internet, 13(5), 111.

12) Ali, T. E., Chong, Y. W., & Manickam, S. (2023). Machine Learning Techniques to

ISSN: 2057-5688

Detect a DDoS Attack in SDN: A Systematic Review. Applied Sciences, 13(5), 3183.

13) Jiang, H., Lin, J., & Kang, H. (2022). FGMD: A robust detector against adversarial attacks in the IoT network. Future Generation Computer Systems, 132, 194-210.

14) He, K., Kim, D. D., & Asghar, M. R. (2023). Adversarial Machine Learning for Network Intrusion Detection Systems: A Comprehensive Survey. IEEE Communications Surveys & Tutorials.

15) Masum, M., Shahriar, H., Haddad, H., Faruk, M. J. H., Valero, M., Khan, M. A., ... & Wu, F. (2021, December). Bayesian hyperparameter optimization for deep neural network-based network intrusion detection. In 2021 IEEE Inter national Conference on Big Data (Big Data) (pp. 5413-5419). IEEE.