IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

Machine Learning Based Intrusion Detection System (IDS)

Prof. Pritam Ahire[1], Abhijeet Mowade[2], Nikhil Bankar[3]

Professor, Computer Engineering, D.Y. Patil Institute of Engineering and Technology Ambi, Pune, India[1]

Student, Computer Engineering, D.Y. Patil Institute of Engineering and Technology Ambi, Pune, India[2]

Student, Computer Engineering, D.Y. Patil Institute of Engineering and Technology Ambi, Pune, India[3]

Abstract: In more recent times, there has been an increase in the number of people using computers, as a result of which there is widespread use of the Internet. The use of the Internet enables hackers to access computers using new, more sophisticated, and more complex forms of attacks, to protect computers from them Intrusion Detection System (IDS) is employed, which has been trained using a number of machine learning techniques as well as datasets. In some networks, the datasets used are acquired over time and usually contain up-to-date data. Furthermore, they are imbalanced and unable to store enough data to withstand all types of attacks. The efficiency of current IDSs is harmed by these inconsistencies and out dated datasets, especially for attacks that are infrequently encountered. We propose a machine learning-based IDSs in this paper, using K-Nearest Neighbour, Decision-Tree, SVM, LSTM, and SMOTE algorithms. To make IDS more logical, an up-to-date security database, CSE-CIC-IDS2018, can be used in place of older and more widely used datasets. The selected database is also not balanced. As a result, utilizing a data model known as the Synthetic Minority Oversampling technique (SMOTE), the rate of inequality in the dataset is lowered to improve the reliability of the system and to avoid inconsistent access and false alarms, a mechanism based on the types of attacks was developed. Data is processed in small classes, and their numbers grow to medium data size in this fashion. The proposed strategy considerably boosts the detection rate of attacks that are infrequently encountered, according to experimental results.

Keywords: IDS, intrusion detection, SVM, LSTM, SMOTE, machine-learning, CSE-CIC-IDS2018.

I. INTRODUCTION

As a result of technological developments, much of the real-world transactions have been made online available through the internet in the cyber world. Therefore, banking, shopping, online examinations, electronic commerce, communication, and many such operations are widely used within the internet. With the widespread use of smartphones, people can connect to this global network and perform transactions at any time and anywhere. While digitalization facilitates regular human activities, networks are frequently assaulted by attackers who take advantage of the Internet's anonymous environment due to server weaknesses and newly developed network intrusion tactics not only to steal certain information or money but also to slow down the performance of network services. Security administrators traditionally choose password protection methods, encryption techniques, and access controls in addition to firewalls as a way to protect the network. However, those methods are insufficient for protecting the system. As a result, many administrators and managers prefer to utilise Intrusion Detection Systems (IDSs) to detect malicious assaults by monitoring network traffic, as seen in Figure 1. Intrusion is described as any illegal activity that compromises the data's confidentiality, availability, or integrity within an information system. IDSs are the most common method of identifying this type of threat. IDSs are mostly preferred means of detecting this type of activity. IDS can be divided into three groups: Signature-based Intrusion Detection Systems (SIDS) Systems, Anomaly-based Intrusion Detection Systems (AIDS), and Hybrid Systems. SIDS keeps signatures of malicious activities on a database and attempts to gain access through pattern matching methods. At the moment, AIDS is trying to learn the normal work ethic and set some as suspicious. In this type of system, there is no need to use a signature-base, and the system can target zero-day attacks that have never been experienced before. Hybrid systems are built on a combination of SIDS and AIDS to increase the rate of acquisition of known risky activities by reducing the negative i.e. false positive rate of zero-day attacks. Because of the benefits of AIDS, many existing IDs directly apply or benefit from AIDS contained by a hybrid approach. These IDSs need to be trained by using a machine learning model to process databases. Many of the functions in this article have adopted older data sets, which contain unwanted details and uneven volumes of data types. While we may encounter some data sets containing up-to-date data, the unequal size of data types remains a challenge for researchers. The effectiveness of the IDS is directly related to the selected learning model and the quality of the data sets used. A good quality database can be defined as a database that develops better performance metrics in a real-world interaction. As mentioned in [1], In the case of unequal division, training for one category (minority) far exceeds the training set of another category (minority), where, a minority class is often the most popular categories [2] the inequality database presents a problem for investigators. The database is said to be unequal when class allocations are unequal [3]. This is a common problem in many classification problems due to the

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

data sets used. An unequal database result in a split that is used in the general category; however, for most of them, the goal is to try to find a minority class [4] this results in a large error in separating the samples of the minority class and the larger goals that can be missed. To maximize data quality, it should be measured in terms of data types. Therefore, in this paper, we aim to use up-to-date datasets to train IDS to develop a knowledge base for the detection of an anomaly. To improve the efficiency of the system, a comparison task was performed using six different machine learning algorithms. To increase the detection rate of low-sample attacks, a data-generating tool is used, and the results obtained from the current work are compared to those of previous tests.

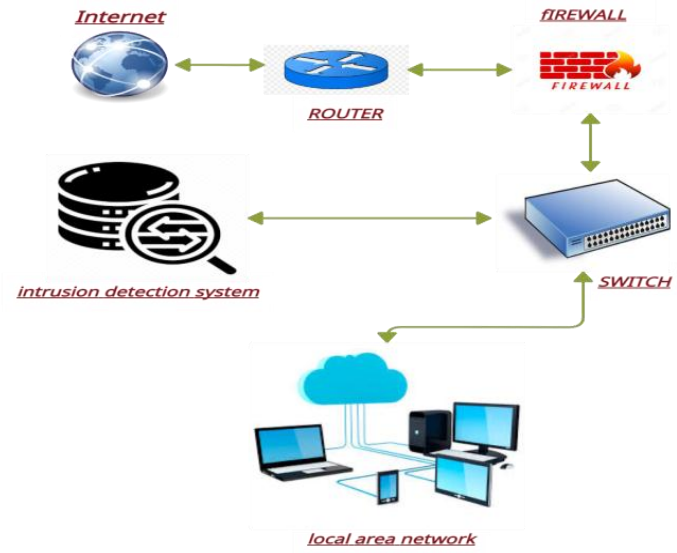

Fig. 1 A sample IDS on Local Area Network

II. EXISTING SYSTEM

Intrusion Detection Systems are striking areas not only for cybersecurity research but also for case studies. In the last few years, many papers have been published on this subject. In this section, these notable pieces of research (especially related to unequal data) are briefly discussed. In 2019, Gao et al. used the NSL-KDD dataset to test and develop IDS using a consistent ensemble learning model [5]. They used four different algorithms K Nearest Neighbor, Deep Neural Networks, Random Forest, and Decision Tree. Also, they designed a compatible voting algorithm. They used the NSL-KDD-Test+ file to verify their path. The decision tree algorithm's accuracy is 84.2 % and the flexible algorithm's final accuracy is 85.2 % . Finally, they compared the relevant research papers and found that the consistency of the results is embedded in their integration model. An online Principal Component Analysis (PCA) study designed to address the problem of misdiagnosis is proposed in [6]. Their approach is focused on using online platforms for major issues. Going through the few categories of the target state by oversampling, their proposed algorithm allows them to determine the randomness of the target state. Comparison between PCA and other acquisition algorithms supported the efficiency, and accuracy of the proposed method. Also, their algorithm has reduced computational costs and memory requirements. Yueai and Junjie proposed a two-phase strategy with a load balancing model (such as an online and offline category) using IDS [7]. In the online section, the system has taken packets from the network and then received attacks. Currently, in the offline category, the training database has been used to create an offline model. They used SMOTE to sample and did their classification with AdaBoost and Random Forest algorithms. Their test results showed that SMOTE and AdaBoost were not working properly. Abdulhammed et al. (2019) used the CIDDS-001 database to manage unequal databases to create effective IDS for a variety of strategies [8]. They have successfully studied the CIDDS-001 sample methods and tested this database by voting, Deep Neural Networks, Variational Autoencoder, Random Forest, and stacking learning algorithms. This program received 99.99% accuracy when using unequal datasets.

III. DATASETS

Researchers can use public datasets or they can use their own datasets. In the following paragraphs, several selected datasets are mentioned and compared with their content and properties.

A. ***KDD CUP99:*** KDD Cup99 was created in 1998 by DARPA to detect network volatility and was used in the 1999 KDD Cup Challenge to test IDS [9], This database is one of the most popular databases in the field of data mining and machine learning. There are about 5 million details in the standard database. About 80% of the data are details of the

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

attack, and the remaining 20% are benign [10]. 41 areas in the database can be grouped under three headings; basic features, traffic features, and content features.

B. ***NSL-KDD:*** The NSL-KDD database was created in 2009 to solve problems related to unfamiliar data in the KDD Cup 99 database [11]. The reliability of the systems developed over the years was questioned, as there was no accurate IDS dataset.

C. ***CIC-IDS2017:*** CIC-IDS2017 was created in 2017 and has the features of most recent and practical attacks in the world that year. It was built by evaluating network traffic using time stamp information, IPs for source and destination, ports for source and destination, attacks and protocols. [12]. 86 network-related features with IP addresses and forms of attacks are included. In accordance with the final database testing framework in 2016, the conditions for establishing a reliable database are determined. Prior to the construction of the CIC-IDS2017 database, no IDS dataset acquisition data met the process of building a reliable database, built-in 2016.

D. ***CSE-CIC-IDS2018:*** The outline concept has been used to create a CSE-CIC-IDS2018 database [13]. The most recent data available in 2018/2019 is the Canadian Institute for Cybersecurity. These profiles can be used by agents or individuals to create events on the network and can be used on different network protocols with different approaches. In addition, the database was developed by analyzing the standards used in constructing CIC-IDS2017. In addition to the basic procedures, it offers the following benefits:
This is one of the recent databases right now. The two profiles were separated, using five methods of attack on the database. The numbers of the benign and attacks are shown in Table II. Also, this table shows IDS Database and its features.

- The number of duplicate data is very low,
- Uncertain information is almost non-existent,
- The database is in CSV format, so it is ready for use without processing.

E. **IMBALANCED RATIO OF KNOWN DATASEETS:** Table I lists the quantity of records in the most popular and widely used datasets, which are divided into classes. These datasets are not balanced, as can be shown, these datasets are not balanced. For accurate calculation of the system's efficiency, this imbalanced structure is needed to be formulated. The imbalance ratio which can be calculated as in Equation 1 can be used as the metric.

$$\boldsymbol{Imbalanced\ Ratio} = \boldsymbol{\rho} = \frac{\boldsymbol{max_i\{C_i\}}}{\boldsymbol{min_i\{C_i\}}} \qquad \text{Equation 1}$$

Where *Ci* shows the data size in the class i. In other words, imbalance ratio can be defined as the fraction between the number of instances of the majority (max) class and the minority (min) class. According to this equation the imbalance ratio of the most popular and recent datasets are listed as in Table II. There is a vast gap between the data classes which also affects the efficiency of the system. Additionally, sophisticated hackers focus on the development of minority data types to reach their targets. Therefore, to increase the efficiency of the system, this imbalance rate should be decreased.

TABLE I DATA SIZE OF DATASETS

Dataset	Class-1	Class-2	Class-3	Class-4	Class-5	Class-6
KDD CUP99	4,113,233	553,301	45,268	18,599	112	-
NSL-KDD	77,054	53,387	14,.077	4,833	119	-
CIC-IDS2017	2,358,036	453,438	15,967	1,966	36	21
CSE-CIC-IDS2018	2,856,035	1,289,544	286,191	93,063	513	53

TABLE III IMBALANCED RATIO OF KNOWN DATASETS

Dataset	Imbalance Ratio
KDD CUP99	***36,725***
NSL-KDD	***684***
CIC-IDS2017	***112,287***
CSE-CIC-IDS2018	***53,887***

IV. PROPOSED SYSTEM

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

Many IDS development studies have been conducted over the years, and increasing detection accuracy is the most critical metric for developers. However, if the dataset is imbalanced and a specific category composes the most significant part of the dataset, then the use of accuracy as a single metric is not much acceptable. If there is a large gap between the data size within the majority and minority categories, sophisticated attackers can focus on minority attack types to increase their efficiency. Therefore, in this paper, we focus on removing the effect of asymmetry between classes in the dataset by increasing the average accuracy of the system. As mentioned before, many current IDSs are developed over Anomaly Detection by identifying the normal data with the use of six machine learning algorithms. As such, many helpful tools have been created over the last few decades, and currently, the Python programming language, as one of the most popular development environments, has become very important for implementing new learning-based systems. The use of new libraries, such as Scikit-Learn (Sklearn) provides excellent flexibility and ease of use not only for system development but also for testing.

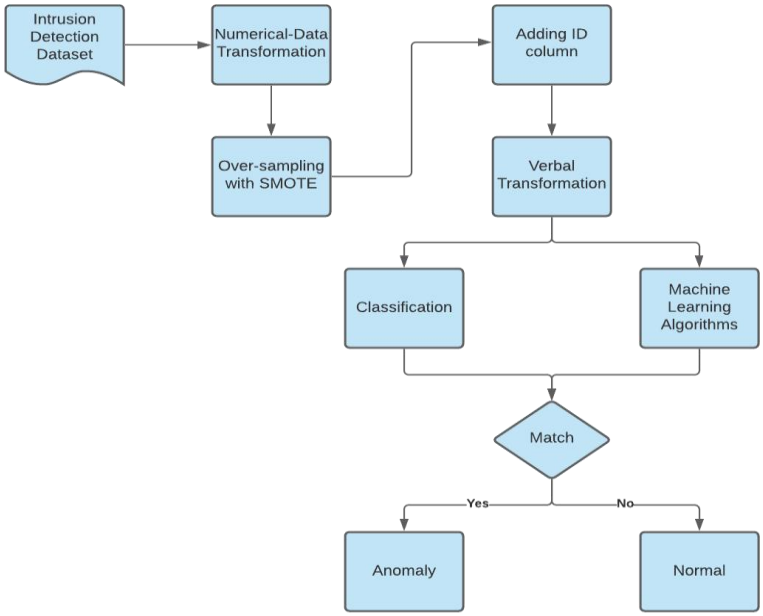

Fig. 2 Flowchart of Proposed System

Our system uses Graphical User Interface (GUI) for efficient use as shown in figure 3. In our system there is a module to insert dataset and to pre-process it by applying numeric transformation and deleting empty entries in dataset, then by Synthetic Minority Oversampling Technique (SMOTE) the imbalanced ratio of dataset is removed and the module is trained with Support Vector Machine and Long Short-Term Memory (LSTM) algorithm. Then by importing test dataset machine is tested and it detects anomaly or normal class of dataset and finally predicts result.

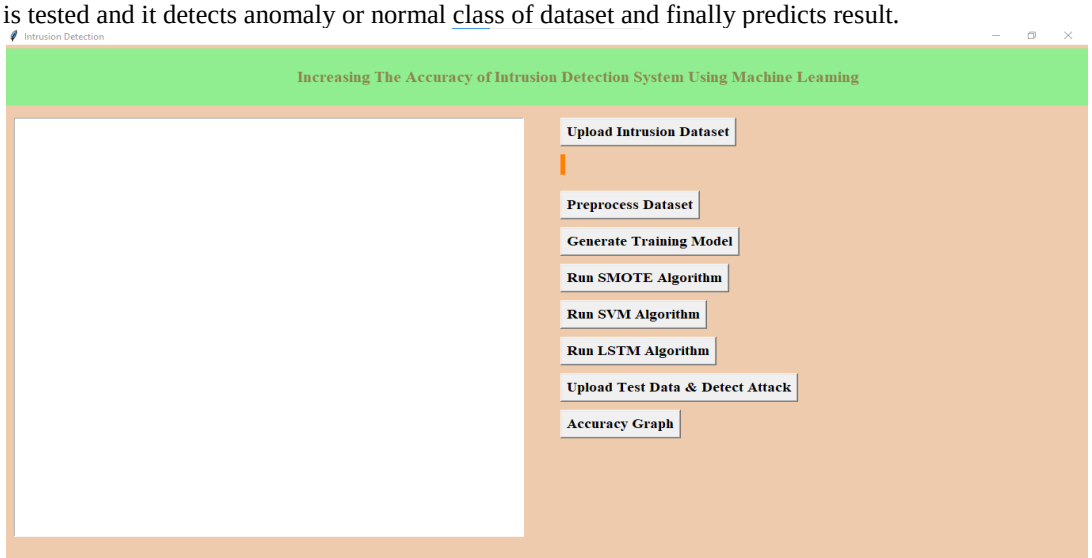

Fig. 3 Graphical User Interface of System

V. WORKING OF SYSTEM

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

As shown in figure 3 in GUI we have a button Upload Intrusion Dataset here we will import our Intrusion detection dataset and then go on to data pre-processing where numerical data transformation takes place then we generate the training and test model by splitting the dataset then by applying Synthetic Minority Oversampling Technique (SMOTE) imbalanced ratio of dataset is removed and further the machine is trained with Support Vector Machine (SVM) and Long Short-Term Memory (LSTM) Machine Learning Algorithms to detect anomaly. Finally test dataset is loaded with Upload Test Dataset Button and attack type is detected whether normal or anomaly, at last result Prediction id done by pressing Accuracy graph button

A. Importing Dataset

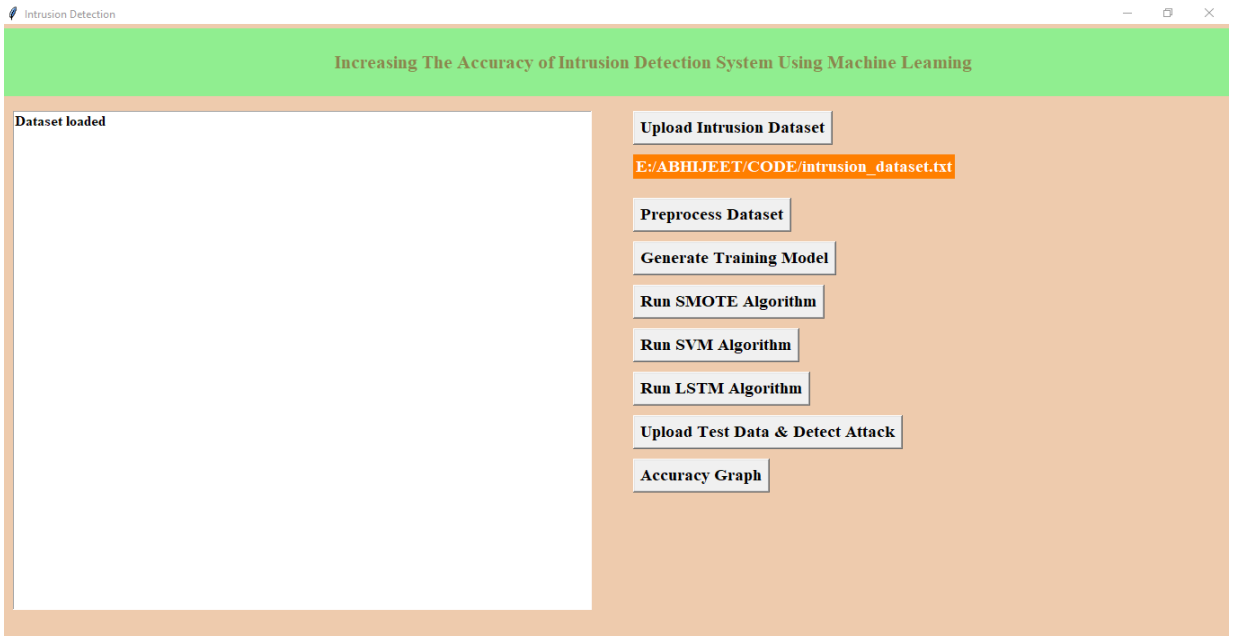

Fig. 4 Importing Intrusion Dataset

B. Pre-processing the Uploaded Dataset

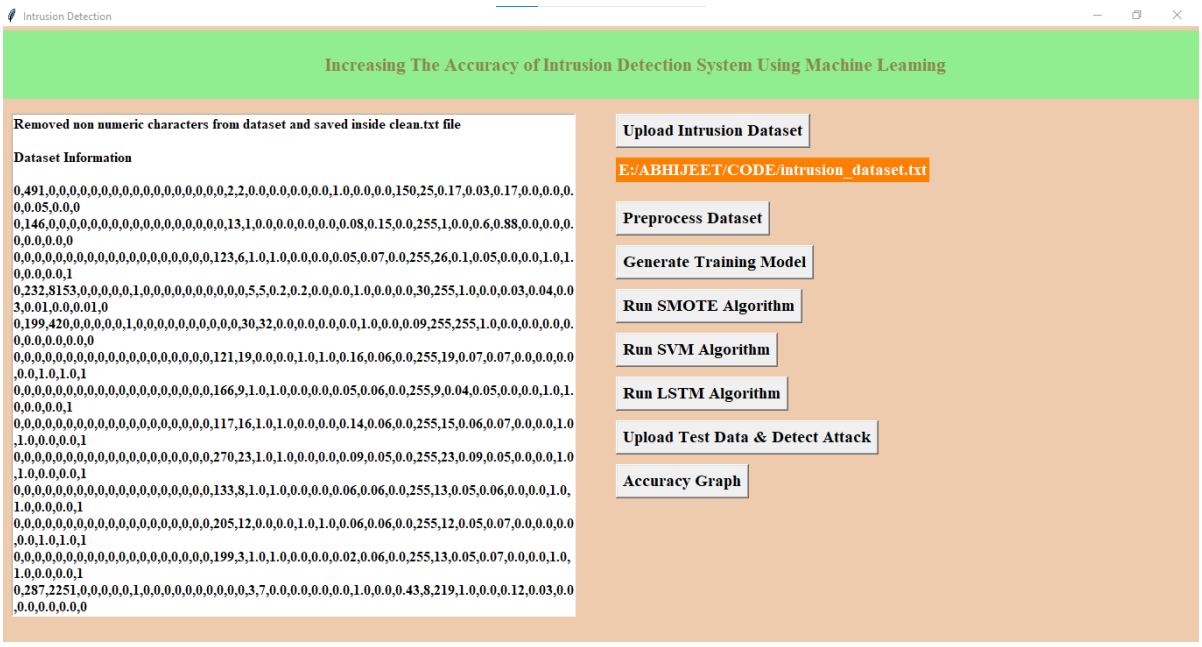

Fig. 5 Pre-processing the Dataset in System

C. Generating training model

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

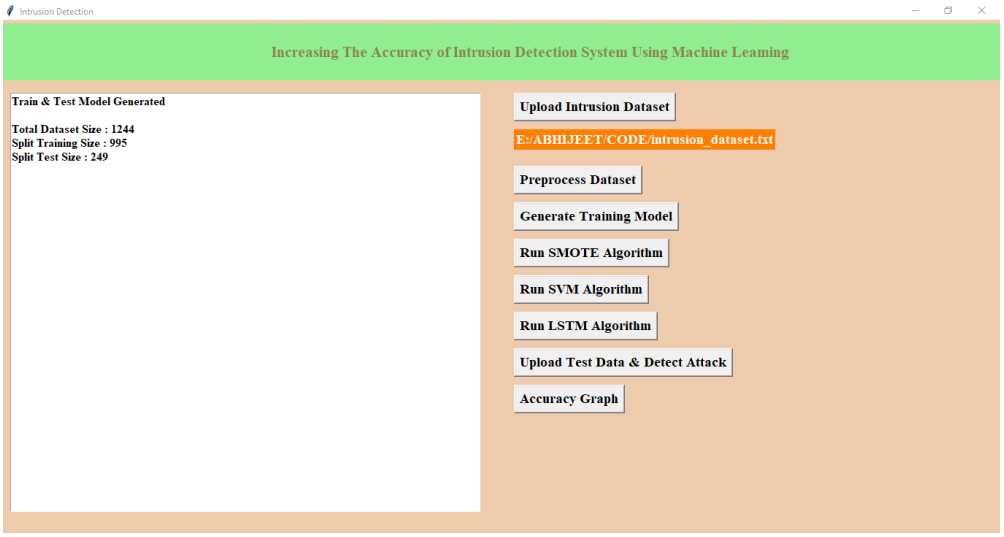

Fig. 6 Splitting the Dataset In Training and Testing Set

D. Applying SMOTE

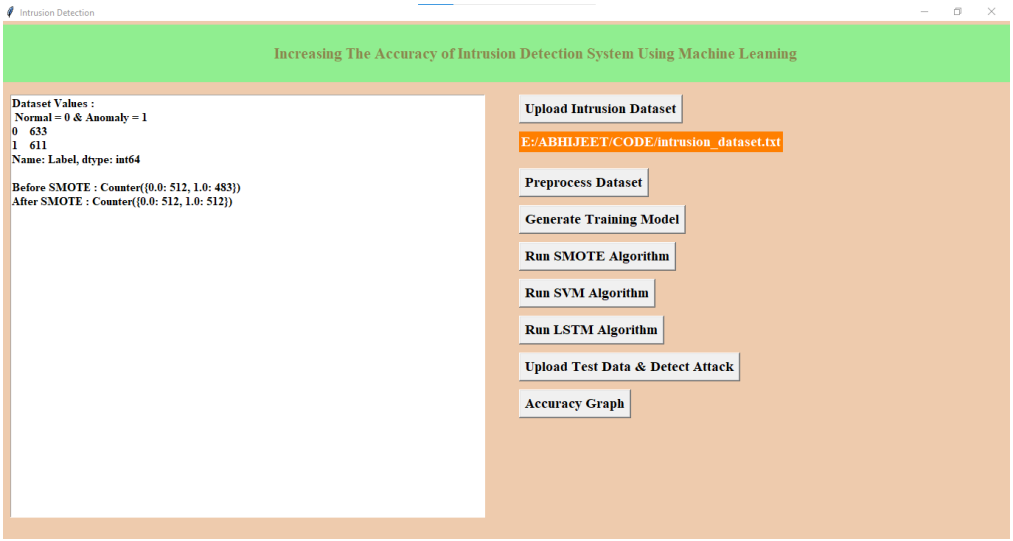

Fig. 7 Application of Synthetic Minority Oversampling Technique Algorithm

E. Training the Machine with SVM and LSTM algorithm

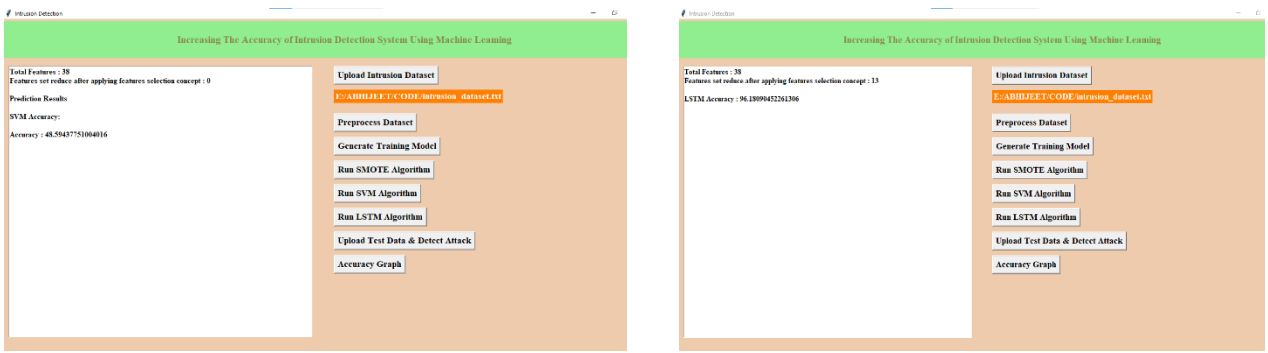

Fig. 8 Training the System with SVM and LSTM Algorithms

F. Attack Detection with Test Data

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

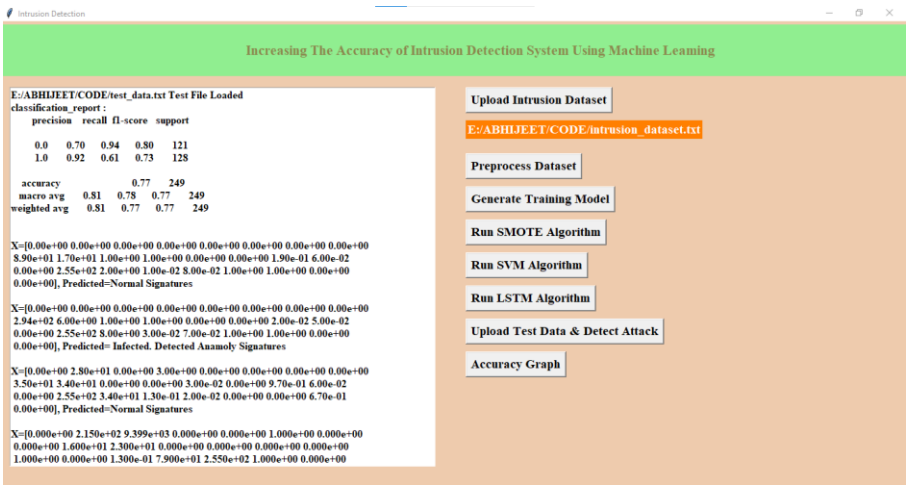

Fig. 9 Attack Detection Using Testing Data

G. Result Prediction Accuracy Graph

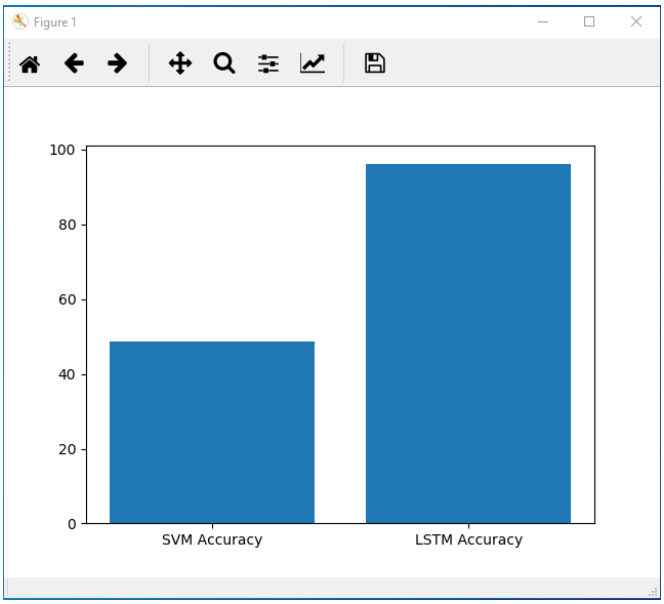

Fig. 10 Accuracy Graph of Result

VI. ALGORITHMS USED

A. Decision Tree

Decision Tree (DT) is one of the supervised learning algorithms used for classifying the numerical and class data. It has a pre-defined goal description. It also has leaf nodes supported by decision-making steps to achieve one of the topdown targets of the algorithm structure [14]. It takes advantage of its simple design to process large amounts of data quickly. The most sophisticated trees may have to contend with database fragmentation in some circumstances. In such cases, decision trees become more difficult, and it is difficult to achieve any goals. Another problem in the decision tree algorithms is overfitting. To fix this difficulty, some leaf nodes from the decision tree are excised. Information gain and entropy should be calculated for decisive trees.

B. Random Forest

Random Forrest (RF) is a type of built-in surveillance system that can be used for regression problems and classification [15]. It is painless to use, and it creates a decision forest through Decision Making and solves a problem in this way. With this, it creates a random collection of trees. During the process, more than one Decision tree is trained to provide the most accurate classes. Most of the time, without using a parameter, it can give really good results. It is one of the most popular methods because it provides instant and accurate results even in mixed, incomplete, and noisy databases.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

C. KNN K-Nearest Neighbour

K Nearest Neighbours (KNN) is a supervised learning algorithm. Unlike other supervised learning algorithms, it does not have a training phase [16]. KNN is implemented using data from the first phase of the sample. The K data is selected, which is the closest neighbour to the new data which must be determined by which sample class to be added. The range of new data to be added to any original sample groups taken from the data showing the K near neighbour property

D. (Support Vector Machine(SVM)

SVM (Support Vector Machine) is a supervised machine learning technique that can be used to solve classification and regression problems. It is, however, mostly employed to solve categorization difficulties. Each data item is represented as a point in n-dimensional space (where n is the number of features), with the value of each feature being the value of the SVM algorithm for a given coordinate. Then we accomplish classification by locating the hyper-plane that clearly distinguishes the two classes.

E. Long Short-Term Memory(LSTM)

Long short-term memory (LSTM) is a deep learning architecture that uses an artificial recurrent neural network (RNN). LSTM has feedback connections, unlike normal feedforward neural networks. It can process not only single data points, but also complete data sequences. For example, activities like unsegmented, connected handwriting identification, speech recognition, and anomaly detection in network traffic or IDSs can all benefit from LSTM (intrusion detection systems).

F. Synthetic Minority Oversampling Technique(SMOTE)

One of the most widely used oversampling approaches to overcome the imbalance problem is SMOTE (synthetic minority oversampling technique). Its goal is to achieve a more balanced distribution of classes by replicating minority class examples at random. SMOTE creates new minority instances by combining existing minorities. For the minority class, it uses linear interpolation to create virtual training records. or each example in the minority class, these synthetic training records are constructed by randomly selecting one or more of the k-nearest neighbours. The data is reconstructed after the oversampling procedure, and many classification models can be applied to the processed data.

VII. EXPERIMENTAL RESULT

The performance of machine learning algorithms in intrusion detection processes is investigated in this work. The mos recent dataset available was used for training and testing. Except for KNN, all of the implemented algorithms have their parameters set by default. The number of classes in the KNN algorithm was discovered to be six (one for non-attack types and five for attack types). To decrease the variability of the performance results due to the random generation of train and test sets, the K-Fold Cross- Validation method was used in the experiments. The chosen Kvaluewas 5, in which the training and test data were divided into 80% to 20%. Proposed systems were implemented in Keras/Tensorfiow using the Python programming language, and Scikit learn libraries. To calculate the performance measure of the proposed systems; *Accuracy*, *Precision*, *Recall*, *F1-Score* and *Error Rate* values are used [17]. These metrics are calculated according to Equations 2- 8.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \qquad \textit{Equation 2}$$

$$Accuracy_i = \frac{TP_i + TN_i}{TP_i + FN_i + FP_i + TN_i} \qquad \textit{Equation 3}$$

$$Average\ Accuracy = \frac{\sum_{i=1}^{l} \frac{TP_i + TN_i}{TP_I + FN_I + FP_i + TN_i}}{l} \qquad \textit{Equation 4}$$

$$Error\ Rate = 1 - Accuracy \qquad \textit{Equation 5}$$

$$Precision = \sum_{i=1}^{l} \frac{TP_i}{TP_i + FP_i} \qquad \textit{Equation 6}$$

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

$$Recall = \sum_{i=1}^{l} \frac{TP_i}{TP_i + FN_i} \qquad \textit{Equation 7}$$

$$F1 - Score = \frac{(\beta^2 + 1)\, Precision * Recall}{\beta^2 Precision + recall} \qquad \textit{Equation 8}$$

where *TPi* is the *ith* True Positive, *FPi* is the *ith* False Positive, *FNi* is the *ith* False Negative, *l* is the number of multiclass, and β is the balancing factor. The most common choice for fi is 1,which is a harmonic mean of precision and recall. The definition used of *accuracy* is critical because accuracy is the most vital metric used to measure the effectiveness of prediction systems. Accuracy often refers to the complete accuracy of the system, However, *Accuracyi* can also refer to an individual accuracy of class i. For an imbalanced dataset, the final definition of accuracy -which is the average of the individual accuracies- is critical for researchers. In this paper, we have implemented six different machine learning algorithms as K Nearest Neighbor, Decision Tree, and Linear Discriminant Analysis, SVM and LSTM. The performance metrics are obtained through the original dataset and extended dataset with sampled data on attack types. As the first metric, the accuracy is measured.

As discussed above, there are some comparisons of the proposed algorithms such as accuracy, time, precision, recall, f1-score. However, to measure the efficiency of a system, a comparison is made between the present study and recent work, (published in 2018) the results of which are depicted in Table III. The present study and the comparison study [18] have one machine-learning algorithm in common (random forest). The use of sampled data leads to, a considerable increase in the accuracy of the system, as 99.34% accuracy rate is measured. The fact that we employ most recent dataset instead of the out-of-date dataset is a significant difference between the two publications. In addition, a comparison of trained IDSs to other machine learning algorithms (e.g., SVM, RBF, and ELM) reveals that the taught IDSs are more efficient. Table III: Table of comparisons (*accuracy figures are estimated and may vary). Additionally a comparison with other machine learning algorithms (i.e. SVM, RBF, and ELM), shows that the trained IDSs are more efficient than these other algorithms.

TABLE III COMPARISON TABLE (*ACCURACY VALUES ARE WRITTEN APPROXIMATELY DEPENDING ON THE REFERENCED PAPER).

Reference [15]		Normal		Sampled	
Algorithm	Accuracy (%)	Algorithm	Accuracy (%)	Algorithm	Accuracy (%)
		ADA	99.69	ADA	99.60
SVM Lin	98.8	DT	99.66	DT	99.57
SVM RBF	98.3	RF	99.21	RF	99.35
RF	97.7	KNN	98.52	KNN	98.58
ELM	99.5	GB	99.11	GB	99.29
		LDA	90.80	LDA	91.18

VIII. CONCLUSION

In recent years, due to the extended use of the Internet, computing devices can connect to a global network at any time and from anywhere. However, the anonymous form of Internet results in lots of security breaches in the network, which results in intrusions. Furthermore, current attackers are more skilled, and they can develop fresh malware with the help of automated production tools, relying on the limited detection capability of Intrusion Detection Systems (IDSs). Pre-collected datasets are commonly used to train IDSs. Almost all of these datasets, however, are unbalanced, with imbalance ratios ranging from 648 to 112,287. Imbalanced datasets result in bias towards the majority class, and in some extraordinary situations, minority classes are ignored. However, these minority classes are generally positive classes. Therefore, the imbalance ratio should be decreased to increase the efficiency of the system and to decrease its average accuracy. To decrease the imbalance-ratio, a data sampling model was used by increasing the data size of the minority groups. The experimental results showed that the implemented models have a very good accuracy level when compared with recent literature The average accuracy of the models increased between 4.01 percent and 30.59 percent when a sampling dataset was employed. Many machine learning applications are being migrated to deep learning models due to the efficiency of big data applications. This paper has been a preliminary study to examine the success of deep learning algorithms in detecting small sample attacks in up to date datasets. Therefore, deep learning algorithms should be used in future work. By using a different design methodology, it is expected that the efficiency of the system will increase.

IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering

Vol. 10, Issue 5, May 2021

DOI 10.17148/IJARCCE.2021.10591

REFERENCES

[1] J. M. Johnson and T. M. Khoshgoftaar, ``Survey on deep learning with class imbalance," *J. Big Data*, vol. 6, no. 1, p. 27, 2019.
[2] A. Ali, S. M. Shamsuddin, and A. L. Ralescu, ``Classification with class imbalance problem:Areview," *Int. J. Adv. Soft Comput. Appl.*, vol. 7, no. 3, pp. 176_204, 2015.
[3] F. Provost, ``Machine learning from imbalanced data sets 101," in *Proc. AAAI Workshop Imbalanced Data Sets*, Menlo Park, CA, USA: AAAI Press, 2000.
[4] S. Barua, M. M. Islam, X. Yao, and K. Murase, ``MWMOTE-majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 405_425, Feb. 2014.
[5] X. Gao, C. Shan, C. Hu, Z. Niu, and Z. Liu, ``An adaptive ensemble machine learning model for intrusion detection," *IEEE Access*, vol. 7, pp. 82512_82521, 2019.
[6] Y.-J. Lee, Y.-R. Yeh, and Y.-C.-F. Wang, ``Anomaly detection via online oversampling principal component analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 7, pp. 1460_1470, Jul. 2013.
[7] Z. Yueai and C. Junjie, ``Application of unbalanced data approach to network intrusion detection," in *Proc. 1st Int.Workshop Database Technol. Appl.*, Apr. 2009, pp. 140_143.
[8] R. Abdulhammed, M. Faezipour, A. Abuzneid, and A. Abumallouh, ``Deep and machine learning approaches for anomaly-based intrusion detection of imbalanced network traffic," *IEEE Sens. Lett.*, vol. 3, no. 1, pp. 1_4, Jan. 2019.
[9] G. Karatas, O. Demir, and O. Koray Sahingoz, ``Deep learning in intrusion detection systems," in *Proc. Int. Congr. Big Data, Deep Learn. Fighting Cyber Terrorism (IBIGDELFT)*, Dec. 2018, pp. 113_116.
[10] G. Karatas and O. K. Sahingoz, ``Neural network based intrusion detection systems with different training functions," in *Proc. 6th Int. Symp. Digit. Forensic Secur. (ISDFS)*, Mar. 2018, pp. 1_6.
[11] M. Tavallaee, E. Bagheri,W. Lu, and A. A. Ghorbani, ``A detailed analysis of the KDD CUP 99 data set," in *Proc. IEEE Symp. Comput. Intell. Secur. Defense Appl.*, Jul. 2009, pp. 1_6.
[12] A. Gharib, I. Sharafaldin, A. H. Lashkari, and A. A. Ghorbani, ``An evaluation framework for intrusion detection dataset," in *Proc. Int. Conf. Inf. Sci. Secur. (ICISS)*, Dec. 2016, pp. 1_6.
[13] I. Sharafaldin, A. Habibi Lashkari, and A. A. Ghorbani, ``Toward generating a newintrusion detection dataset and intrusion traffic characterization," in *Proc. 4th Int. Conf. Inf. Syst. Secur. Privacy*, 2018, pp. 108_116.
[14] N. Frosst and G. Hinton, ``Distilling a neural network into a soft decision tree," 2017, *arXiv:1711.09784*. [Online]. Available: http://arxiv.org/abs/1711.09784
[15] M. Belgiu and L. Dr guμ, ``Random forest in remote sensing: A review of applications and future directions," *ISPRS J. Photogram. Remote Sens.*, vol. 114, pp. 24_31, Apr. 2016.
[16] S. Zhang, X. Li, M. Zong, X. Zhu, and R. Wang, ``Efficient kNN classification with different numbers of nearest neighbors," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 5, pp. 1774_1785, May 2018.
[17] M. Sokolova and G. Lapalme, ``A systematic analysis of performance measures for classification tasks," *Inf. Process. Manage.*, vol. 45, no. 4, pp. 427_437, Jul. 2009.
[18] I. Ahmad, M. Basheri, M. J. Iqbal, and A. Rahim, ``Performance comparison of support vector machine, random forest, and extreme learning machine for intrusion detection," *IEEE Access*, vol. 6, pp. 33789_33795, 2018.