SLUJST
Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Models Comparison Based On Intrusion Detection Using Machine Learning

Maimuna Yusuf Ma'aji[1*] and Muhammad Sirajo Aliyu[2]
[1]Department of Computer Science, Umaru Musa Yar'adua University Katsina, Nigeria
[2]Department of Computer Science, Faculty of Computing, Federal University Dutse, Jigawa State
meimuselite@gmail.com[1*], msirajoa@fud.edu.ng[2]

[*]Corresponding Author

Abstract

Network security is the greatest challenges in our current generation. Network intrusion detection (NID) is one of the fundamental techniques used to protect computer networks from threads. However, there is contemplation on the possibility and sustainability of the traditional approaches employed especially with the current modern networks. Recently, majority of the researchers employ the application of machine learning models such as logistic regression (LR), random forest (RF), K-nearest neighbor (KNN) and Support Vector Machines (SVM) among others to address the NID problem. In this research, the Network intrusion classification system has been analyzed based on KDD Cup '99 data set. In the classifier implementation section, the three most widely used models were employed; K-nearest neighbor, random forest and logistic regression as our classification algorithms. The attack detection accuracy and training time was used to evaluate the models. The random forest provides the best detection accuracy of 99.5% and 74 second training time provide by logistic regression.

Keywords: *Data Mining, Intrusion Detection, KDD CUP 99, Network security*

1. Introduction

The Network Intrusion Detection System (NIDS) has been facing a serious challenge with the evolution of modern technologies which makes it lack effective and robust ways of intrusion detection. A lot of solutions operate using less-capable and signature-based techniques, as opposed to anomaly detection techniques. There are many reasons for this reluctance to switch, which includes the increase in false error rate, lack of availability of reliable training data, long and complexity of training data and dynamics behavior of the system. The present situation will attain a point in which reliance on such methods leads to ineffective and inaccurate detection. This current challenge is to give a room to develop or find a widely-accepted anomaly detection method capable of handling limitations occurring in modern networks. We are concerned with three main limitations, which contribute to this network security challenge. For instance, behavioral changes need to be easily attributable to specific elements of a network, e.g., operating system versions, individual users or protocols. Another cause is the diversity of data traversing through modern networks and number of different protocols. This can possibly be the most significant challenge and introduces high-levels of complexity and difficulty when trying to differentiate between abnormal and normal behavior. It increases the difficulty in widens the scope for potential exploitation and establishing an accurate norm. Recently, the main focus within NIDS research has been the application of machine learning techniques such as Decision Trees, Naive Bayes, Random Forest, Logistic Regression, K-nearest Neighbor and Support Vector Machines (SVM).

Machine learning (ML) received much attention in the recent years. There has been a current development in this area as researchers now focuses in providing more transparency in the algorithms. Artificial Intelligence (AI) has made lot things possible which once were thought as a tedious task, ML being one of the domains of AI was considered to be so important and influential. ML is completely based on statistics and mathematics. It has been a technique to

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

build intelligent systems. Similar to AI, it assists in the forecasting future by analyzing past data and experience. There have been so many applications of ML like classification, object differentiation, text classification, speech recognition, checking of destroyed crops, prediction of weather, medical diagnostics and face recognition etc. (Wenke and Stolfo 2000; Zheng et al. 20001; Mohammed et al. 2009). ML works purely on data set, especially Big Data that played a vital role in its evolution (Duan and Xiao 2018; Christine et al. 2009; Foster and Tom 2000).

Intrusion detection system (IDS) is a technique or a program that tries to find indications that the computer has compromised (Moradi and Zulkernine 2003). The system attempts to detect/identify an intruder breaking into computer system or authenticated user misuses system resources. Intrusion detection is very important and has captured the attention of researchers, security professionals and network administrators. It is the art of detecting inappropriate, unauthorized, or anomalous activity on computer systems. IDSs are classified as host based, network based, or application based depending on their mode of data used for analysis and deployment (Arman et al. 2008). In addition, IDSs can also be classified as anomaly based or signature based depending upon the attack identification or detection technique. The signature-based systems are trained by extracting specific signatures or patterns from previously known attacks while the anomaly-based systems learn from the normal data collected when there is no anomalous activity (Wang et al. 2005; Wang et al. 2016). The main aim of IDS is to identify as many attacks as possible with a smaller number of false alarms. i.e., the system should have high detecting accuracy. Moreover, accurate technique that cannot manage large volume of network traffic and has slow running power in decision making could not serve the purpose of an intrusion detection system (Gupta et al. 2010). The main issue in IDs is presence of large number of false alerts; this issue has motivated many researchers to discover the solution for reducing false alerts.

It has been reported by Wang and Zheng (2018) that the simplest and more reliable machine learning technique in detecting intrusion are; Logistic Regression (LR), Random Forest (RF) and K-nearest Neighbor (KNN). Therefore, this motivated us to compare the three most prominent machines learning models; LR, KNN and RF to evaluate their performances using KDD CUP 99 data set, these models are very capable of correctly analyzing a wide-range of network traffic to detect and classify false alerts in intrusion detection. The main aim of this research is to compare the three methods and obtain excellent procedures to classify KDD CUP 99 dataset in order to have an easiest, high accuracy and less training time for knowing the best procedure so as to provide and recommend the most efficient technique to the researchers for future study on KDD CUP 99 dataset.

2. Related Works

Warrender et al. (1999) developed many intrusions detection technique based on system call trace data. They tested a technique that employed sliding windows to determine a database of normal sequences to form a database for testing against test instances and detect/classify instances according to those in the normal sequence database. This involves the maintenance of huge database of normal system call trace sequences. Wenke et al. (2000) developed a Mining Audit Data for Automated Intrusion Detection models. It is among the best-known data mining technique in intrusion detection.

Agarwal et al. (2000) developed a two-stage general-to-specific framework for learning a rule-based model (PN rule) to learn classifier techniques on KDD CUP 99 data set. Daniel et al. (2001) developed Audit Data Analysis and Mining, which is an intrusion detector construct to identify intrusions using data mining methods. In the first stage it absorbs training data that is free of attacks. In the second stage, it employed an algorithm to group attacks, false alarms and unknown behavior.

Abraham (2001) developed a technique using Data Mining for Intrusion Detection, which is a real-time NIDS for anomaly detection and misuse. It uses Meta rules, association rules and characteristic rules. It utilizes data mining technique to provide a description of network data. Zheng et al. (2001) developed a statistical neural network classifier for anomaly detection, the technique identifies User Datagram Protocol (UDP) flood attacks. It also compared different neural network classifiers, in which the Back Propagation Neural Network (BPNN) has found to be more efficient in developing IDS. Xu (2006) proposed a framework based on machine learning for adaptive intrusion detection. Mrutyunjaya and RanjanPatra (2009) present a study based on the performance of three data mining classifier

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

technique, J48, ID3 and Naïve Bayes are evaluated on the KDD CUP 99 data set. Mohammed et al. (2009) presented a comprehensive analysis classification used to predict the extent of attacks over the network. They compared zero R classifier, Decision table classifier and Random Forest classifier with KDD CUP 99 databases from MIT Lincoln laboratory.

Chihab et al. (2013) present works which involve five data mining algorithms to make a comparison between them which uses network intrusions and get the best proposal of a hybrid classifier based on random forest and naive Bayes algorithms. The results indicate that the hybrid system improved the prediction with reduced, consuming time.

Keerthika and Priya (2015) developed a procedure that focuses on the naive feature reduction, in addition to feature selection methods such as information gain and gain ratio for reducing the redundant and irrelevant of features. This proposal used naive Bayes classifier to design intrusion detection system. Tesfahun and Bhaskari (2015) present an effective and reliable hybrid layered intrusion detection system by combining misuse and anomaly IDS for detecting both previously known and unknown attacks.

Aggarwal and Sharma (2015) assessed several classification algorithms like Naive Bayes, C4.5, Random Forest and Decision Table. They compared these classification algorithms in WEKA with KDD CUP 99 dataset. The classifiers were resolved according to metrics such as precision, accuracy, and F-score. Random Tree presents the best output aggregate in contrast to the algorithms, which provide high detection and low false alarm rate. Mukund and Nayak (2016) present the current existing algorithms for intrusion detection system to use Hadoop Distributed File System (HDFS) in order to reduce the false alarm rate, in their work they used decision tree method and enhanced it in the process with the multi-system capabilities of the HDFS. Their approach reduced the time taken by the DFS and makes a significant improvement on the accuracy of the IDS. Gupta et al. (2016) proposed IDSs to monitors the network and forbidden access to devices. IDSs plays a significant role as it protect the data's features and integrity. The proposed IDSs employ NSL-KDD dataset to learn the manner of the attacks based on the methods of data mining such as K-means clustering and logistic regression. Hence, it creates rules for classifying network activities. The results indicates that linear regression provides the highest accuracy of 80% in detecting attacks while the K-means clustering showed 67%. Kabir et al. (2017) presented a novel technique for intrusion detection system based on sampling with Least Square Support Vector Machine (LSSVM).

Duan and Xiao (2018) present a huge volume of the data and unbalanced dataset, intrusion data were inevitable obstacles. So, solve those issues utilizing the fuzzy c-means technique to reconstruct feature vectors according to central points. Wang (2018) present deep learning procedure for intrusion detection which implemented in GPU-enabled TensorFlow and evaluated utilizing KDD CUP 99 dataset. Raheem and Saleh (2018) in their work provide a technique for intrusion detection according to tree calculation on the KDD CUP-99.

2.1 Theoretical Review

In this research three (3) classification models (LR, RF and KNN) were examine base on KDD CUP 99 data set. There are a lot of evaluation metrics for Intrusion Detection. The following is a description of some of these metrics (Chen et al. 1996). The three models were evaluated base on the metrics defined below.

1. True Positive (TP) - Attack data that is correctly classified as an attack.
2. False Positive (FP) - Normal data that is incorrectly classified as an attack.
3. True Negative (TN) - Normal data that is correctly classified as normal.
4. False Negative (FN) - Attack data that is incorrectly classified as normal.

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

2.1.1 KDD CUP 99 data set

The KDD CUP 1999 dataset derived from 1998 DARPA Intrusion detection Evaluation program. It is one of the most widespread datasets collected over a period of nine weeks for a LAN simulating a typical U.S. Air Force

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

LAN. The dataset contains a collection of simulated raw TCP dump data, in which multiple intrusions attacks was introduced and widely used in the research community from seven weeks of network traffic. The dataset contains 311,029 unlabeled and 4,898,430 labeled connection records. The labeled connection records consist of 41 attributes, which include basic features (e.g., protocol type, packet size), Domain knowledge features (e.g., number of failed logins) and timed observation features (e.g., % of connections with SYN errors). The KDD CUP 99 dataset represents feature values of a class, where each class is categorized and labelled either normal or attack. The classes in the dataset are categorized into one normal class and four main intrusion classes (Christine et al. 2009):

- *Normal*: connections are generated by simulating user behavior.
- *DoS attacks*: use of resources or services is denied to authorize users.
- *Probe attacks:* information about the system is exposed to unauthorized entities.
- *Privilege attacks:* access to account types of administrators is gained by unauthorized entities.
- *Access attacks:* access to hosts is gained by unauthorized entities.

The attack classifications

- **dos_attacks**=['apache2','back','land','neptune','mailbomb','pod','processtable','smurf','teardrop','udpstorm','worm']
- **probe_attacks**=['ipsweep','mscan','nmap','portsweep','saint','satan']
- **privilege_attacks**=['buffer_overflow','loadmdoule','perl','ps','rootkit','sqlattack','xterm']
- **access_attacks**=['ftp_write','guess_passwd','http_tunnel','imap','multihop','named','phf','sendmail','snmpgetattack','snmpguess','spy','warezclient','warezmaster','xclock','xsnoop']

2.1.2 Logistic Regression

Logistic regression (LR) is one of the best algorithms to start with classification algorithms. It uses a logistic function to construct binary output model. The output of the LR will be a probability ($0 \leq x \leq 1$), and can be used to forecast the binary 0 or 1 as the output (if $x < 0.5$, output= 0, else output=1).

LR acts very similar to linear regression. It also calculates the linear output, followed by a stashing function over the regression output. Sigmoid function is the frequently used logistic function. You can see below clearly, that the z value is same as that of the linear regression.

$$z = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \cdots \quad (2)$$

$$h(\theta) = g(z) \quad (3)$$

$$g(z) = \frac{1}{1+e^{-z}} \quad (4)$$

The $h(\theta)$ corresponds to $P(y = 1|x)$, is the probability of output to be binary 1, given input x. $P(y = 0|x)$ will be equal to $1 - h(\theta)$.

For instance, when value of z is 0, $g(z)$ will be 0.5. Whenever z is positive, $h(\theta)$ will be greater than 0.5 and output will be binary 1. Likewise, whenever z is negative, value of y will be 0. By using a linear equation to find the classifier, the output model also will be a linear one, that means it splits the input dimension into two spaces with all points in one space corresponds to same label.

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

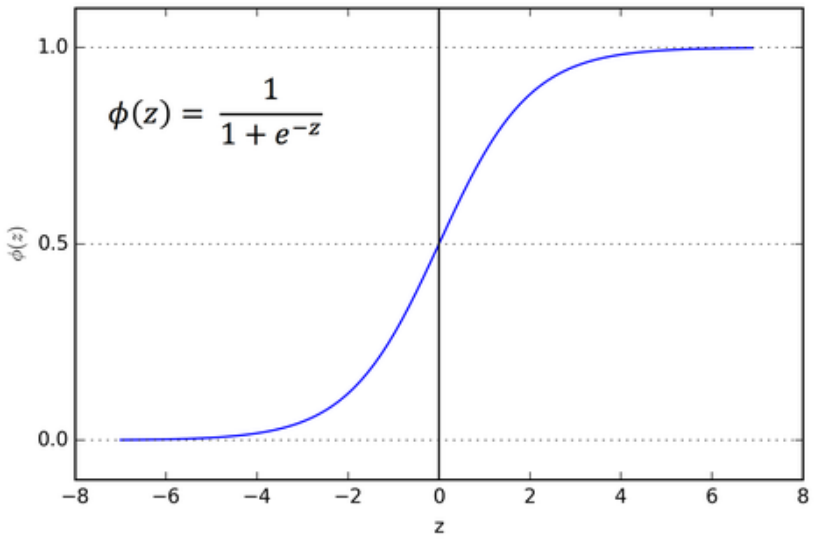

Figure 1: The distribution of a sigmoid function.

Loss function: the mean squared error cannot be used as loss function (like linear regression), because we use a non-linear sigmoid function at the end. MSE function may introduce local minimums and will affect the gradient descend algorithm.

So, we use cross entropy as our loss function here. Two equations will be used, corresponding to y=1 and y=0. The basic logic here is that, whenever my prediction is badly wrong, (e.g.: y' =1 & y = 0), cost will be $-log(0)$ which is infinity.

$$J(\theta) = \frac{1}{m}\sum cost(y', y) \quad (5)$$

$$cost(y', y) = -\log(1 - y') \ if \ y = 0 \quad (6)$$

$$cost(y', y) = -\log(y') \ if \ y = 1 \quad (7)$$

The above equation, stands for training data size, y' stands for predicted output and y stands for actual output.

2.1.3 K-nearest neighbors (KNN)

K-nearest neighbors is used for classification and regression. It is one of the easiest and simplest ML techniques. It is a lazy learning model, with local approximation. The logic behind KNN is to explore neighborhood, assume the test data point to be similar to them and derive the output. KNN look for k neighbors and come up with the prediction.

In KNN classification, a majority voting is applied over the k-nearest data points whereas, in KNN regression, mean of k nearest data points is computed as the output. As a rule of thumb, it selects odd numbers as k. KNN is a lazy learning model where the computations happen only runtime.

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

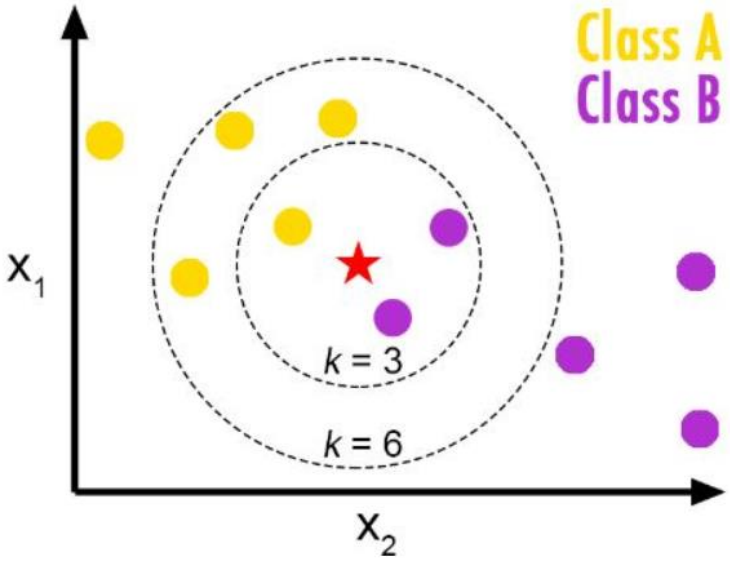

Figure 2: Classification in KNN

Figure 2 above shows yellow and violet points corresponds to Class A and Class B in training data. The red star points to the test data which is to be classified. when k = 3, we predict class B as the output and when K=6, we predict class A as the output.

Loss function: There is no training involved in KNN. During testing, k-neighbors with minimum distance, will take part in classification.

2.1.4 Random Forest (RF)

Random Forest is a popular machine learning algorithm that belongs to the supervised learning technique. It can be used for both Regression problems and Classification in ML. It is based on the concept of ensemble learning, which is a process of combining multiple classifiers to solve a complex problem and to improve the performance of the model.

Random Forest is a classifier that has a number of decision trees on various subsets of the given dataset and takes the mean to increase the predictive accuracy of that dataset. Instead of relying on one decision tree, the random forest takes the forecast from each tree and based on the majority votes of predictions, and it predicts the final output. The greater number of trees in the forest leads to higher accuracy and prevents the problem of over fitting.

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

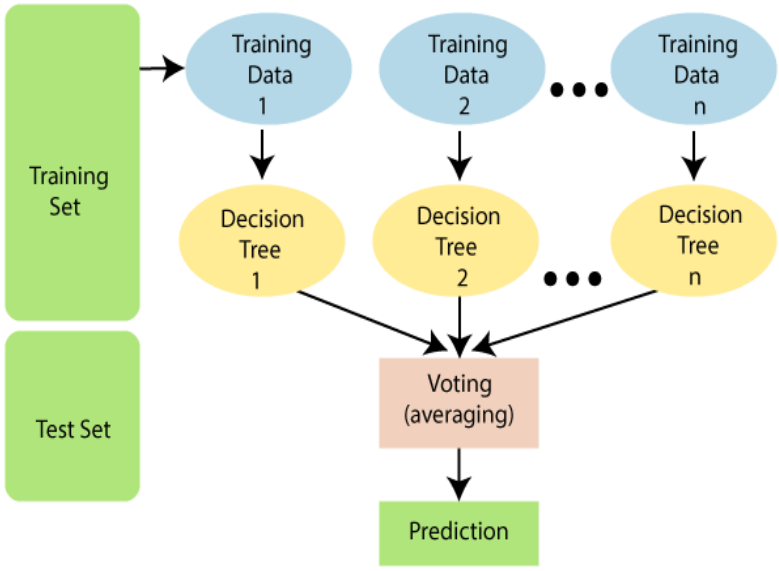

Figure 3: The diagram explains the working of the Random Forest algorithm

Random Forest works in two-phase first is to create the random forest by combining N decision tree, and second is to make forecast for each tree created in the first phase.

The step can be explained in the below steps:
Step-1: Select random K data points from the training set.
Step-2: Build the decision trees associated with the selected data points (Subsets).
Step-3: Choose the number N for decision trees that you want to build.
Step-4: Repeat Step 1 & 2
Step-5: For new data points, find the predictions of each decision tree, and assign the new data points to the category that wins the majority votes.

3. Results

The purpose of this research is to compare three methods and obtain excellent procedures to classify KDD CUP 99 dataset in order to have an easiest and high accuracy and training time for knowing the best procedure. Transmission Control Protocol (TCP), Internet Control Message Protocol (ICMP) and user datagram protocol (UDP) are the protocol type used in this research.

Table 1 shows the comparison of the three models base on attack detection accuracy and training time. The random forest provides the best detection accuracy of 99.5% and 74 second training time provide by logistic regression. Similarly, figure 4 also shows the box plot comparison of the models in which random forest shows the highest accuracy.

Table 1: Comparison of the three models

Classifier	Accuracy	Training Time in Second
Logistic Regression	39.74	**74**
Random Forest	**99.52**	443
K-nearest Neighbor	98.13	352

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

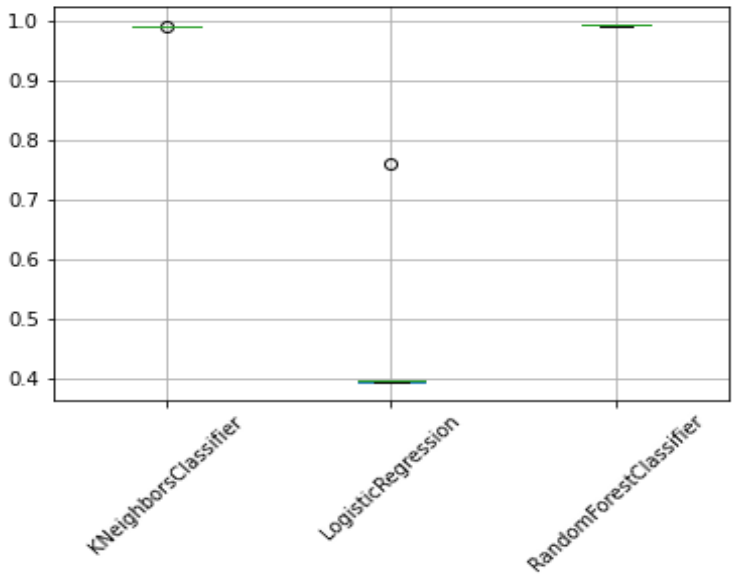

Figure. 4: Box plot of the three models compared (KNN, LR and RF)

The result provide in Table 2 shows the number of attacks base on protocols where all attacks map to 1 and normal map to 0, the 'TCP' was reported to have the highest number of attacks (102,688) followed by 'UDP' with 14,990 number of attack and least among those reported here is 'ICMP' with 8,291 number of attacks. Figure 5 shows a pie chart of the entire attack type base on protocol, in which 'IPSWEEP' is the highest for ICPM, 'normal' is the highest for TCP and UDP.

Figure 6 and 7 also present pictorial representation of count of each flag for attack and normal traffic and also count of each service for attack and normal traffic respectively.

Table 2: Number of attack base on protocol

Attack	Protocol Type		
	ICMP	TCP	UDP
back	0	956	0
buffer_overflow	0	30	0
ftp_write	0	8	0
guess_passwd	0	53	0
imap	0	11	0
ipsweep	3117	482	0
land	0	18	0
loadmodule	0	9	0
multihop	0	7	0
neptune	0	41214	0
nmap	981	265	247
normal	1309	53599	12434
perl	0	3	0
phf	0	4	0
pod	201	0	0
portsweep	5	2926	0
rootkit	0	7	3
satan	32	2184	1417
smurf	2646	0	0

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

spy	0	2	0
teardrop	0	0	892
warezclient	0	890	0
warezmaster	0	20	0
Total	**8,291**	**102,688**	**14,990**

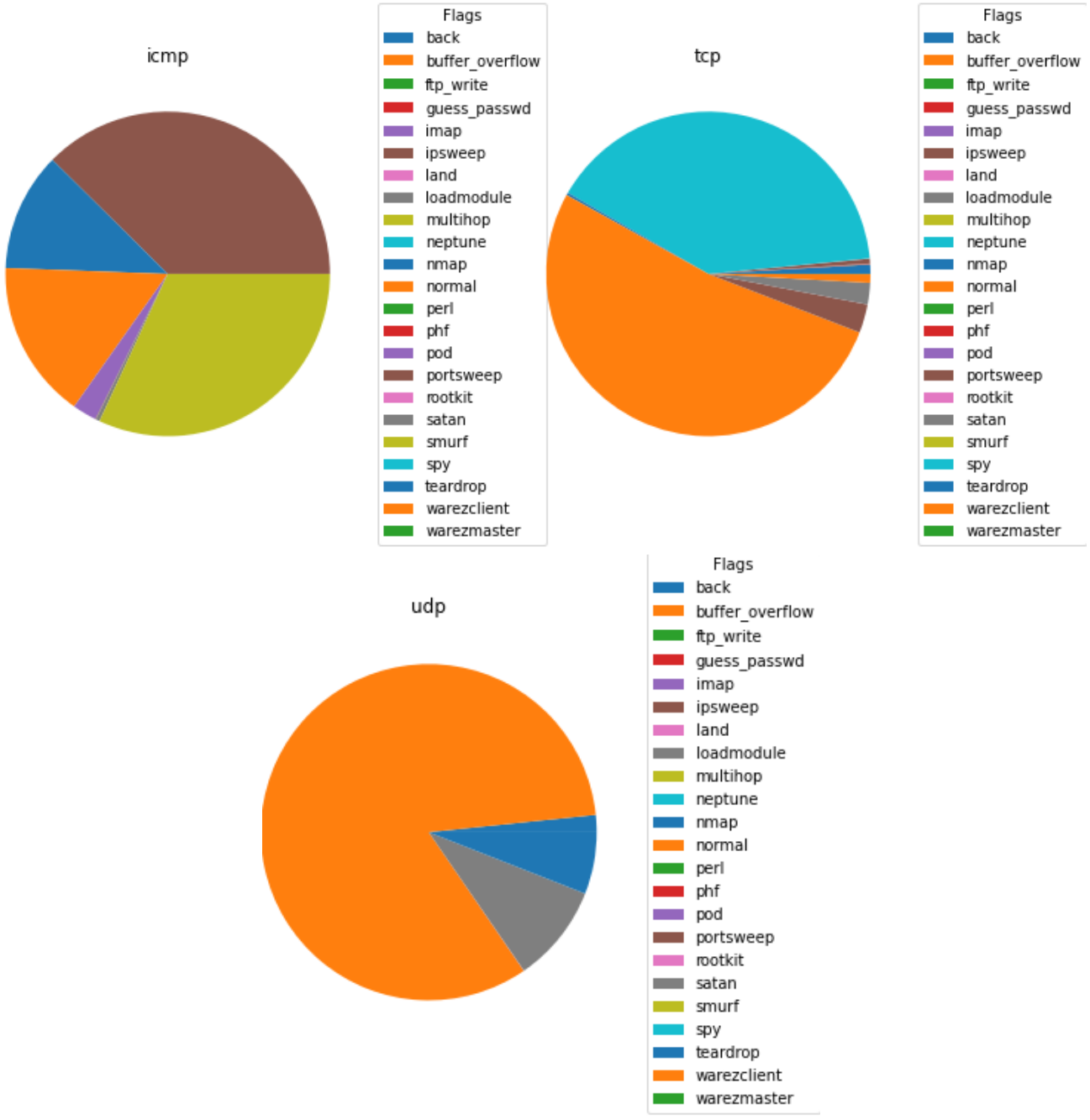

Figure 5: Flag for attack base on three protocols (ICMP, TCP and UDP)

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

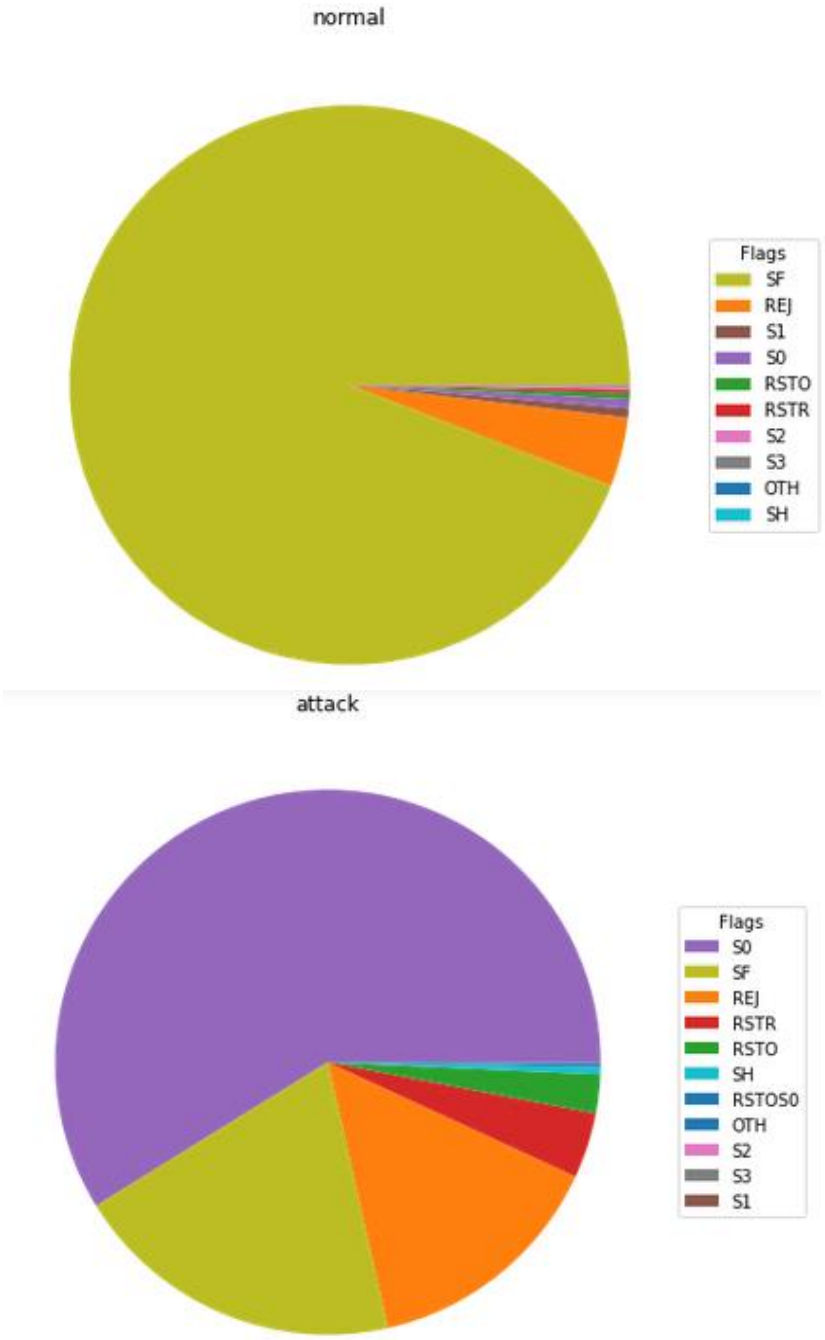

Figure 6: Count of each flag for attack and normal traffic

Sule Lamido University Journal of Science & Technology Vol. 6 No. 1&2 [March, 2023], pp. 74-86
https://doi.org/10.56471/slujst.v6i.358
Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

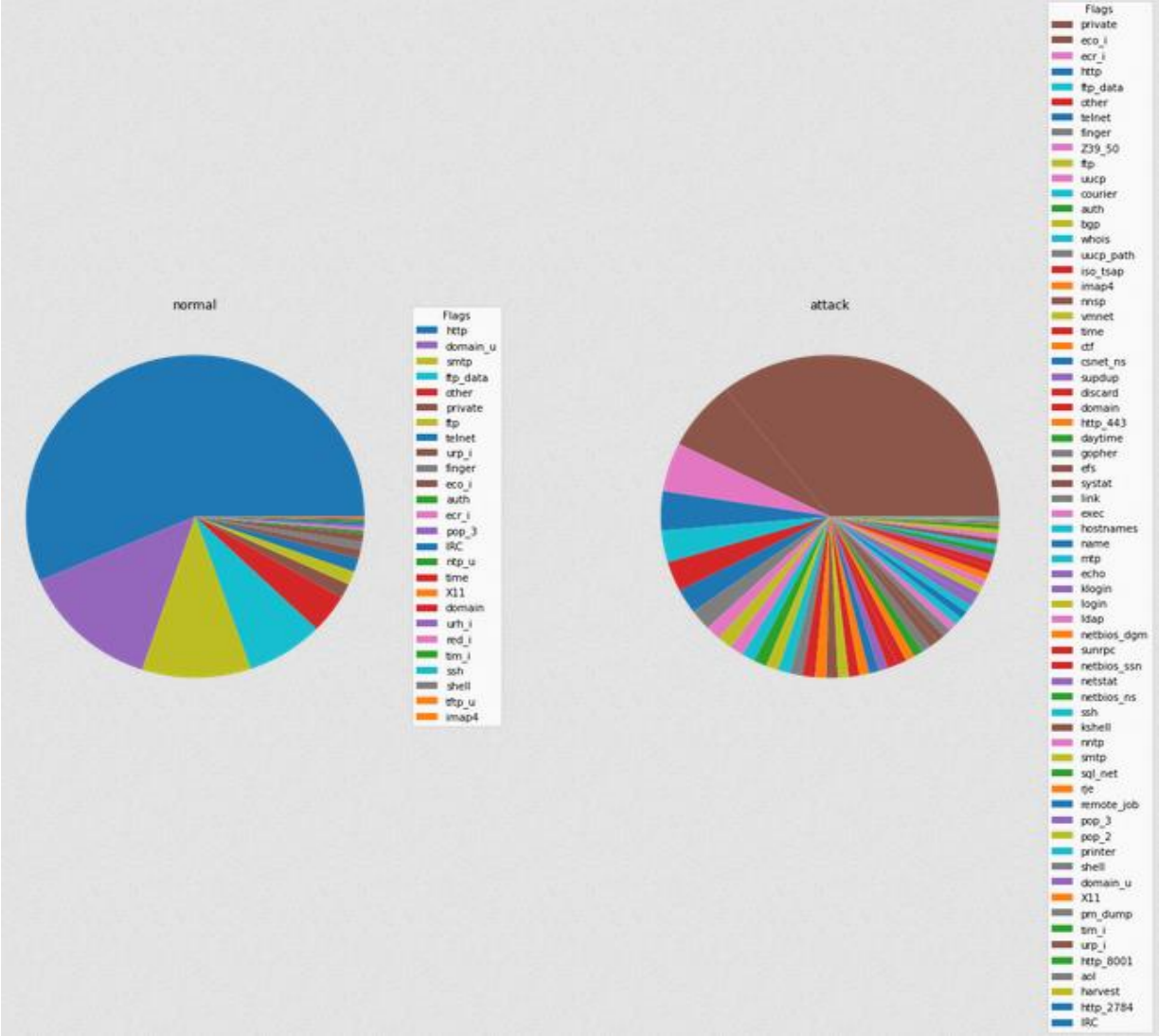

Figure 7: Count of each service for attack and normal traffic

4. Conclusion

There is contemplation regarding the feasibility and sustainability of the approaches employed especially with the current modern networks. Network intrusion detection (NID) is a one of the fundamental techniques use to protect computer networks from threads. This research employs three procedures; logistic regression (LR), random forest (RF) and K-nearest neighbor (KNN) to analyze KDD Cup'99 data set and compare the three models base on attack detection accuracy and training time. The random forest provides the best detection accuracy of 99.5% and 74 second training time provide by logistic regression. Therefore, we recommend random forest for analyzing KDD Cup'99 data set.

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

References

Abraham T. (2001). IDDM: Intrusion Detection Using Data Mining Techniques, Technical report DSTO electronics and surveillance research laboratory, Salisbury, Australia.

Agarwal R., Joshi M. and V. (2000). A New Framework for Learning Classifier Models in Data Mining", Tech. Report, Dept. of Computer Science, University of Minnesota.

Aggarwal P. and Sharma S.K. (2015). An Empirical Comparison of Classifiers to Analyze Intrusion Detection, Proc. of Fifth International Conference an Advanced Computing and Communication Technologies.

Akashdeep Sharma ,Ishfaq Manzoor, Neeraj Kumar, (2017). A Feature Reduced Intrusion Detection System Using ANN Classifier, Expert Systems With Applications

Arman Tajbakhsh, Mohammad Rahmati, and Abdolreza Mirzaei (2008) Intrusion detection using fuzzy association rules, *Applied Soft Computing ASOC509*, Elsevier B.V.

Chen M.S., Han J and Yu Philip S. (1996). Data Mining: An Overview from a Database Perspective, *IEEE Transactions on Knowledge and Data Engineering*, vol.8, No.6, pp.866-883.

Chihab Y., Ouhman A., Erritali m. and Ouahidi B. (2013)," Detection & Classification of Internet Intrusion Based on the Combination of Random Forest and Naïve Bayes, *International Journal of Engineering and Technology (IJET).*

Christine Dartigue, Hyun IK Jang and Wenjun Zeng (2009). A New data-mining based approach for network Intrusion detection, *Proc. of Seventh Annual Communication Networks and Services Research Conference*, pp.372-377.

Daniel B., Couto J., Jajodia S. and Wu N. (2001). ADAM: A Test Bed for Exploring the Use of Data Mining in Intrusion Detection", SIGMOD, vol30, no.4, pp: 15-24.

Duan L. and Xiao Y. (2018). An Intrusion Detection Model Based on Fuzzy C-means Algorithm, *8th International Conference on Electronics Information and Emergency Communication (ICEIEC),* Beijing, pp. 120-123.

Foster Provost and Tom Fawcett (2000). Robust Classification for Imprecise Environment, pp.1-38, Kluwer Academic Publishers.

Gupta KK, Nath B. and Kotagiri R. (2010). Layered Approach Using Conditional Random Fields for Intrusion Detection," *IEEE Transactions on Dependable and Secure Computing*, vol. 7, no. 1, pp. 35–49.

Gupta D., Singhal S, Malik S. and Singha. (2016). Network intrusion detection system using various datamining techniques, *IEEE publication*.

Kabir, E., Hu, J. Wang H. and Zhuo G. (2017). A novel statistical technique for intrusion detection systems, Future Generation Computer Systems

Keerthika G. and Priya D. S. (2015). Feature Subset Evaluation and Classification using Naive Bayes Classifier, *Journal of Network Communications and Emerging Technologies (JNCET)* Volume 1, Issue 1.

Li Y. and Guo L. (2007). An Active Learning Based TCM-KNN Algorithm for Supervised Network Intrusion Detection", *In: 26th Computers and Security*, pp: 459–467.

Moradi M. and Zulkernine M. (2003), A Neural Network Based System for Intrusion Detection and Classification of Attack, *Natural Science and Engineering Research Council* Canada (NSERC).

Mohammed M., Mazid M., Shawkat A. and Kevin S. (2009). Tickle, A Comparison Between Rule Based and Association Rule Mining Algorithms, *Third International Conference on Network and System Security*.

Mukund Y. and Nayak S. (2016). Improving false alarm rate in intrusion detection systems using Hadoop', 21-24 Sept, *International Conference*. Vol.3

Mrutyunjaya P. and Ranjan Patra M. (2009). Evaluating Machine Learning Algorithms for Detecting Network Intrusions, *International Journal of Recent Trends in Engineering*, vol. 1, no.1.

Raheem Esraa and Saleh Alomari (2018). An Adaptive Intrusion Detection System by using Decision Tree OsamahAdil, *Journal of AL-Qadisiyah for computer science and mathematics* Vol.10 No.2

Sathyabama S., Irfan Ahmed M., Saravanan A. (2011). Network Intrusion Detection Using Clustering: A Data Mining Approach, *International Journal of Computer Application* (0975-8887), vol. 30, no. 4.

Tesfahun A. and Bhaskari L. (2015). Effective Hybrid Intrusion Detection System: A Layered Approach, *IJCNIS*, vol.7, no.3, pp.35-41.

Wang, Zheng. (2018). Deep learning-based intrusion detection with adversaries. *IEEE Access* 6, 38367- 38384.

Online ISSN: 2736-0903
Print ISSN: 2736-089X
www.slujst.edu.ng

Wang H., Cao J. and Zhang Y. (2005). A flexible payment scheme and its role-based access control, *IEEE Transactions on knowledge and Data Engineering*, vo. 17, no. 3, 425–436.

Wang D., Zhang Z., Wang P., Yan J and Huang X. (2016). Targeted Online Password Guessing: An Underestimated Threat, *ACM Conference on Computer and Communications Security*, pp. 1242-1254.

Warrender C., Forrest S. and Pearl M. (1999). Detecting Intrusions Using System Calls: Alternative Data Models", in *IEEE symposium on security and privacy*, pp:133-145.

Wenke L., Stolfo S. and J. (2000). A Framework for Constructing Features and Models for Intrusion Detection Systems, *ACM transactions on Information and system security* (TISSEC), vol.3, Issue 4.

Xu X., (2006). Adaptive Intrusion Detection Based on Machine Learning: Feature Extraction, Classifier Construction and sequential Pattern Prediction. *International Journal of Web Services Practices* 2(1-2), pp:49–58.

Zheng Z., Li J., Manikapoulos C.N., Jorgenson J. and ucles J. (2001). HIDE: A Hierarchical Network Intrusion Detection System Using Statistical Pre-Processing and Neural Network Classification, *IEEE workshop proceedings on Information assurance and security*, pp: 85-90.