2021 International Conference on Technologies and Applications of Artificial Intelligence (TAAI) | 978-1-6654-0825-7/21/$31.00 ©2021 IEEE | DOI: 10.1109/TAAI54685.2021.00057

Machine Learning in Cyber Security Analytics using NSL-KDD Dataset

Rui-Fong Hong
Department of Computer Science & Information Engineering, Chaoyang University of Technology
Taichung, Taiwan, R.O.C.
s10927609@cyut.edu.tw

Shih-Cheng Horng
Department of Computer Science & Information Engineering, Chaoyang University of Technology
Taichung, Taiwan, R.O.C.
schong@cyut.edu.tw

Shieh-Shing Lin
Department of Electrical Engineering
St. John's University
Taipei, Taiwan, R.O.C.
sslin@mail.sju.edu.tw

***Abstract*—Classification is the procedure to recognize, understand, as well as group ideas and objects into given categories. Classification techniques adopt training data patterns to predict the likelihood that subsequent data will classify into one of the given categories. Classification techniques utilize a variety of algorithms to classify future datasets through training data patterns. In current society, many network attacks continue to carry out various types of attacks. This work performs data pre-processing and uses Python with machine learning algorithms to analyze the NSL-KDD data set. We use various machine learning methods, such as decision trees, random forests, Naïve Bayes, KNN, Gradient Boosted Trees, and SVM to analyze the confusion matrix and predict the accuracy. We also draw the ROC curve and the AUC area. We calculate the ACC accuracy and make a simple assessment of the quality of different algorithms. Test results show that through data pre-processing, machine learning algorithms can be performed with extremely high accuracy.**

Keywords—Python, Cyber security, Machine learning, Classification, NSL-KDD.

I. INTRODUCTION

In today's society, millions of people rely on various smart devices to connect to the Internet, and there are many dangers in them. The problem of cyber-attacks has penetrated all societies. Many organizations have spent a lot of effort to strengthen cybersecurity. For example, the FBI continues to confirm which units are under cyber-attacks, collect and analyze evidence, and share information about hacking incidents. These organizations said that cyber-attacks are continuing, which is why cybersecurity is important. In the past, cyber threats were much simpler, mainly defined by their technology. Now relying on more advanced network and data infrastructure, the attack surface and the impact of threats are growing day by day.

Figure 1 illustrates that a series of significant attacks on E-commerce websites also mark 2019 when Magecart Group 12 and FIN6 infected thousands of online stores to steal customer credit information. The aforementioned threats highlight the security vulnerabilities in the technology used today. These also illustrate how trends and weaknesses in the industry, equipment, or platform affect the attack pattern [1]. Artificial Intelligence has been around since the 1950s. Artificial intelligence means that computers are as smart as humans. But there was no way to do it at the time.

After the 1980s, there were machine learning methods. Is to make the machine have the ability to learn. When faced with these problems, a huge amount of data needs to be processed, relying on a kind of artificial intelligence technology, machine learning. Machine learning is very similar to the way that humans learn. To allow computers to learn like humans, they usually first classify, then analyze, understand and judge, and finally take action.

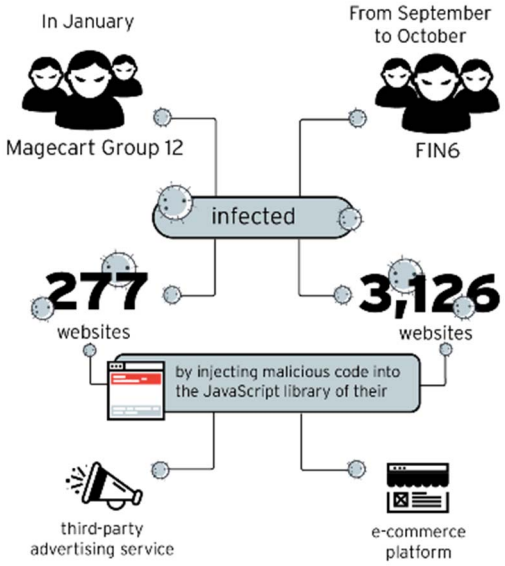

Fig. 1 Magecart Group 12 and FIN6 in 2019.

Classification is the procedure to recognize, understand, as well as group ideas and objects into given categories. Classification techniques adopt training data patterns to predict

2376-6824/21/$31.00 ©2021 IEEE
DOI 10.1109/TAAI54685.2021.00057

the likelihood that subsequent data will classify into one of the given categories. Classification techniques utilize a variety of algorithms to classify future datasets through training data patterns.

The remainder of this paper is divided into five sections. In Section II, we introduce the NSL-KDD data set. In Section III, we introduce the data preprocessing and machine learning algorithms. In Section IV, we demonstrate the experimental results. In Section V, we make a conclusion.

II. NSL-KDD DATA SET

The NSL-KDD data set is a proposed data set to solve some inherent problems of the KDDCUP99 data set mentioned in [2]. NSL-KDD dataset can be downloaded from the Canadian Institute for Cybersecurity (CIC), which is based at the University of New Brunswick [3]. The NSL-KDD dataset deletes duplicate data in KDDcup99 [4] to enhance the accuracy of the data. The data set used in this article is a subset of NSL-KDD, KDDTrain+.TXT, and KDDTest+.TXT, and both data include the tag of the attack type and the difficulty level of the CSV format.

In Table I, we can see that there are four types of attacks in the NSL-KDD data, A denial-of-Service attack (Dos), User to Root Attack (U2R), Remote to Local Attack (R2L), and Probing Attack.

TABLE I. NSL-KDD DATA SET

Dataset	Number of Records					
	Total	Normal	DoS	Probe	U2R	R2L
KDDTrain+ .txt	125972	67342 (53.45%)	45927 (36.46%)	11656 (9.26%)	52 (0.04%)	995 (0.79%)
KDDTest+ .txt	22543	9810 (43.51%)	7458 (33.09%)	2421 (10.73%)	200 (0.89%)	2654 (11.78%)

III. MACHINE LEARNING ALGORITHMS

Artificial intelligence (AI) refers to machines or systems which can simulate human intelligence. AI is often discussed together, and sometimes these two words can be used in common, but they do not refer to the same thing. Machine learning focuses on building a system that learns from data or improves performance through the data accessed.

Now machine learning technology has been widely used in our lives. Machine learning plays a very important role when we transact with banks, shop online, and use social media. This technology brings us a more effective, smooth, and safe experience. Machine learning and other related technologies are developing rapidly, and we are only now beginning to touch the tip of the iceberg. Machine Learning means looking for a Function from Data. First defines a set of functions, second goodness of function, last pick the best function. There are many techniques in Intrusion Detection System (IDS) [5].

A. MACHINE LEARNING

Machine learning is mainly divided into four learning methods [6].

1) Supervised learning: There is a label in the data to determine errors when outputting. This scheme is used to learn the function of a project or find the relation between input and output.

2) Unsupervised learning: There is no label in the data, the machine classifies itself by looking for the characteristics of the data. In future predictions, the features classified by the machine are used to identify, but the identification results may not be correct.

3) Semi-supervised learning: There are a few labels in the data and a lot of them are without labels. The computer only needs to find the features through the data with labels and classify other data. This method can make predictions with higher accuracy and is the most common method.

4) Reinforcement learning: The machine learns through every interaction with the environment to obtain the maximum prediction model. Using reinforcement is a way of learning. In the data without labels, the correct step can be taken. Afterward, the machine can correct itself based on the feedback and finally get the correct result.

B. ALGORITHMS

A free software for the Python programming language is used to check how to use the data to predict network anomalies or attacks, and gradually start using the functions provided in "Scikit-learn". It used to be called "scikits.learn" was also called "sklearn". It is characterized by various classifications including Classification, Confusion Matrix, Accuracy, Support Vector Machine, Gradient Boost Tree, K-Nearest Neighbor, Random Forest, Decision tree, and Navis Bytes [7].

C. DATA Pre-processing

Next, we will perform data extraction. Before throwing the data into the model, we usually have to do some preprocessing of the data, because the amount of data for machine learning is usually very large. Doing some sorting or calculations beforehand can increase the effectiveness or speed of training, and it can also make people more intuitive to Look at the characteristics of the data. How to deal with it depends on the characteristics of the data. When there are non-numerical or unclear features in the data, we may need to convert the data in advance so that the model can be effectively trained. In Table II we will see that some Columns are not included in the original data set [8]-[9].

The first transformation for data transformation is around the attack area. We will first add a column that encodes the "normal" value as 0 and any other value as 1. Use it as a classifier for a simple binary model to identify any attack. As you can see in Table III, we will classify each attack into 'Denial of Service attacks', 'Probe attacks', 'Privilege escalation attack's, and 'Remote access attacks' according to the attack type to obtain more refined predictions. model.

TABLE II. ADD THE COLUMNS LABELS

No.	Columns Name	No.	Columns Name
1	duration	23	count
2	protocol_type	24	srv_count
3	service	25	serror_rate
4	flag	26	srv_serror_rate
5	src_bytes	27	rerror_rate
6	dst_bytes	28	srv_rerror_rate
7	land	29	same_srv_rate
8	wrong_fragment	30	diff_srv_rate
9	urgent	31	srv_diff_host_rate
10	hot	32	dst_host_count
11	num_failed_logins	33	dst_host_srv_count
12	logged_in	34	dst_host_same_srv_rate
13	num_compromised	35	dst_host_diff_srv_rate
14	root_shell	36	dst_host_same_src_port_rate
15	su_attempted	37	dst_host_srv_diff_host_rate
16	num_root	38	dst_host_serror_rate
17	num_file_creations	39	dst_host_srv_serror_rate
18	num_shells	40	dst_host_rerror_rate
19	num_access_files	41	dst_host_srv_rerror_rate
20	num_outbound_cmds	42	attack
21	is_host_login	43	level
22	is_guest_login		

TABLE III. ADD THE COLUMNS LABELS

Attacks Class	Denial of Service attacks	Probe attacks	Privilege escalation attacks	Remote access attacks
Attacks Types	apache2	ipsweep	buffer_overflow	ftp_write
	back	mscan	loadmdoule	guess_passwd
	land	nmap	perl	http_tunnel
	neptune	portsweep	ps	imap
	mailbomb	saint	rootkit	multihop
	pod	satan	sqlattack	named
	processtable		xterm	phf
	smurf			sendmail
	teardrop			snmpgetattack
	udpstorm			snmpguess
	worm			spy
				warezclient
				warezmaster
				xclock
				xsnoop

Feature engineering allows us to delve into the construction of some functions. The items are ‘protocol_type’, ‘service’, and ‘flag’. Through these differences, a basic level of authentication can be obtained. Then add basic digital data "duration", "src_bytes" and "dst_bytes". These can be easily obtained from the current network equipment and can describe in detail what is happening on the network. Then we used 'pd.get_dummies', which is a method that enables us to block hot-encode columns. This will take the found values in a single column and create a separate column for each value, with 0 or 1 indicating whether the column is 'hot'. Then we continue to set the classification target, and conduct two kinds of training for binary and multi-classification.

IV. EXPERIMENTAL RESULTS

A. CONFUSION MATRIX

To understand the predicted data results, a confusion matrix is used in Table IV to observe the predicted and true values. It can be seen that 0 means normal and 1 means attacked. To analyze different machine learning ‘True-negative’, ‘False-negative’, ‘True-positive’, and ‘False-positive’ [10].

B. ALGORITHMS ACCURACY

Model fitting data found that Decision Tree SVM, Random Forest, Gradient Boosting, KNN, and Naive Bayes were used through machine learning. Perform machine learning on the pre-processed data. In Table V, it can be seen that through data pre-processing, other than Naive Bayes, another accuracy has a good performance. The best performance is Random Forest, which shows that Random Forest utilized several trees to construct more effective prediction models.

1) Accuracy: Combine machine learning algorithms to predict accuracy as formula 1.

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{1}$$

C. ALGORITHMS Recall&F1 Score

After understanding the confusion matrix, you can calculate various ratios based on TN, FP, FN, TP to measure the effectiveness of the model. The relevant formulas are listed as follows [11]:

1) Precision: That is, several of the positive samples are predicted correctly as formula 2.

$$Precision = \frac{TP}{TP + FP} \tag{2}$$

2) Recall: That is, how many of the true samples are predicted to be correct as formula 3.

$$Recall = \frac{TP}{TP + FN} \tag{3}$$

3) F1 Score: That is, the harmonic average of precision rate and recall rate as formula 4.

$$F1\,Score = \frac{2}{\left(\frac{1}{Precision}\right) + \left(\frac{1}{Recall}\right)} \tag{4}$$

TABLE IV. EACH ALGORITHM CONFUSION MATRIX

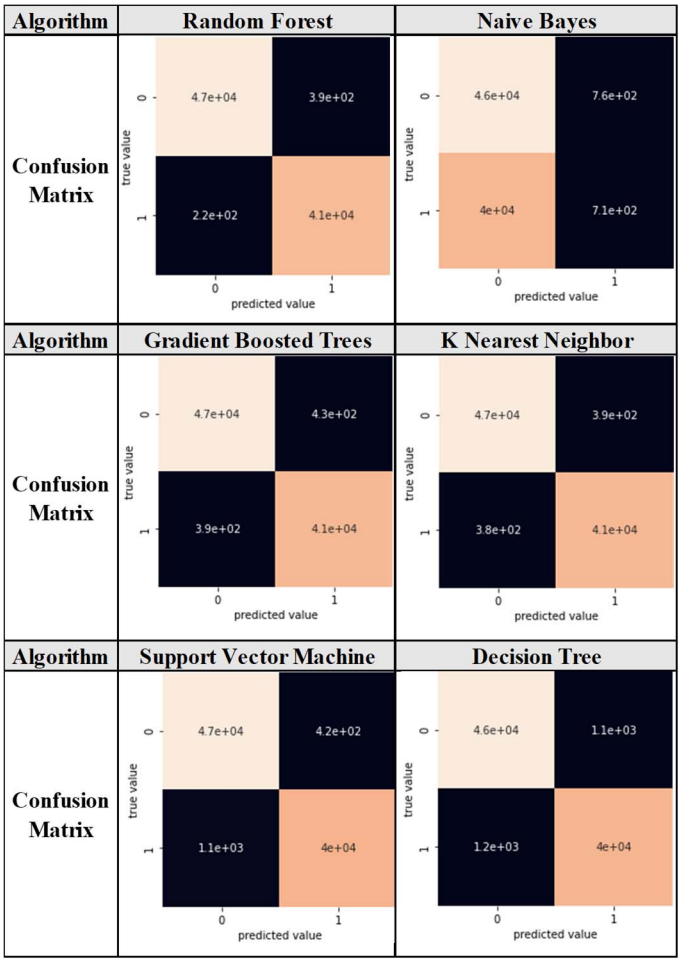

TABLE V. THE ACCURACY OF SIX MACHNE LEARNING ALGORITHMS.

Machine Learning Algorithm	Accuracy
Decision Tree	0.974
Random Forest	0.993
Naïve Bayes	0.534
Support Vector Machine(SVM)	0.982
Gradient Boosted Trees	0.990
K Nearest Neighbor	0.991

The results are shown in Table VI. Under normal circumstances, we only need to use the accuracy rate to measure the performance of the model, which is the "prediction correct rate", that is, the number of guesses (TP+TN) / the total number of samples. However, be careful when the target variables of the training data are not balanced.

TABLE VI. THE PRECISION&RECALL&F1 SCORE ON KDDTEST+DATASET

Machine Learning Algorithm	Precision	Recall	F1 Score
Decision Tree	0.974	0.971	0.972
Random Forest	0.993	0.994	0.992
Naïve Bayes	0.534	0.017	0.033
Support Vector Machine	0.982	0.972	0.980
Gradient Boosted Trees	0.990	0.990	0.989
K Nearest Neighbor	0.991	0.990	0.990

D. ALGORITHMS ROC&AUC

In classification models, the ROC curve and AUC value are usually used as indicators to measure the degree of model fit. Recently, the ROC curve of the model needs to be produced during the modeling process. First, we introduce the principle of Python making a ROC curve. In "sklearn.metrics", there has two functions, "ROC curve" and "AUC". The points on the ROC curve are mainly calculated by these two functions. The ROC diagrams of each machine learning are shown from fig. 2 to fig. 7.

Using the ROC curve to compare the higher risk prediction, you can see that the results show that except for the overlapping phenomenon in Naive Bayes, which indicates that the prediction results are poor, the others have good results. FPR is used as the horizontal axis of coordinates, TPR is used as the number axis, the black one is the random prediction line corresponding to the straight line from [0,0] to [1,1], and the orange one is the machine learning prediction line. As shown in the figure, when the line crosses the random prediction line, the closer it is to the upper left corner, the higher the detection rate and the lower the error detection rate. [0,1] point is the most perfect state. The closer to the lower right corner, the lower the algorithm quality. When the orange line can see the AUC, and when the AUC area is larger, the algorithm is better.

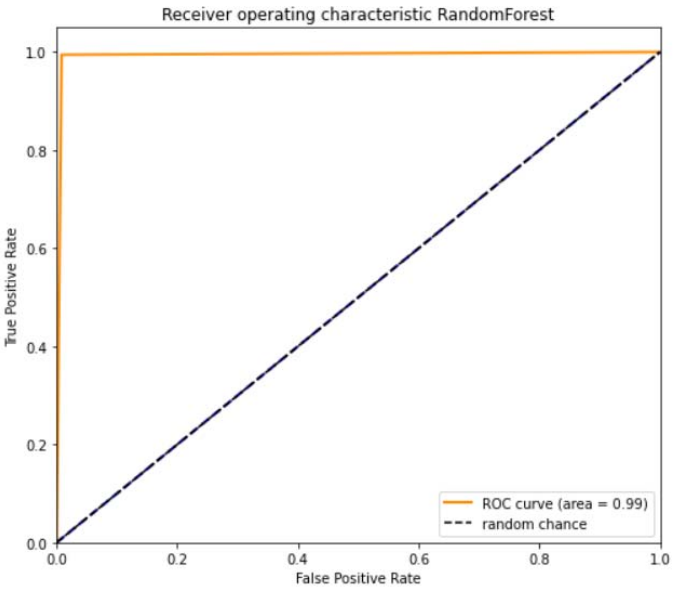

Fig. 2 ROC Random Forest.

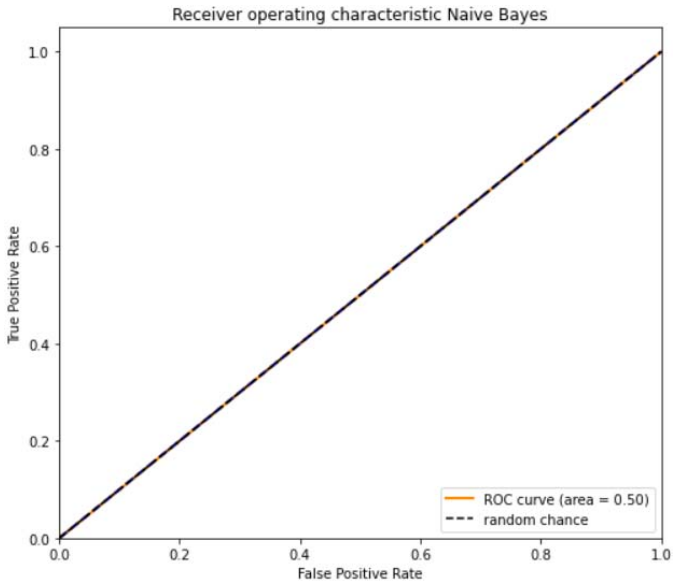

Fig. 3 ROC Naïve Bayes.

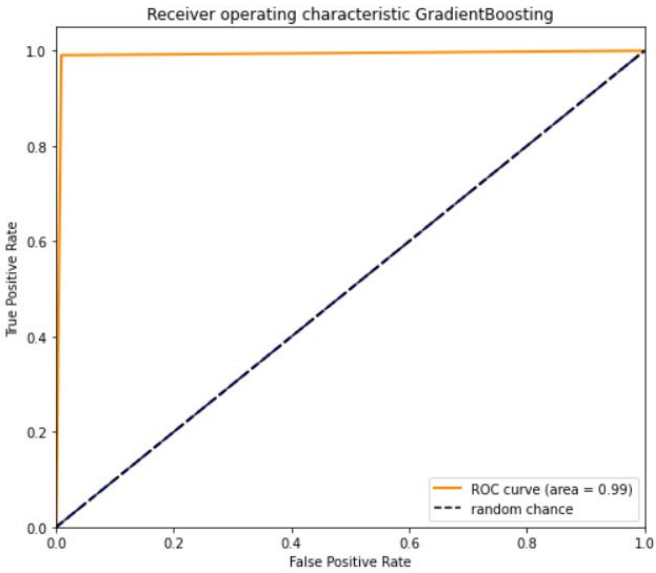

Fig. 5 ROC Gradient Boosting.

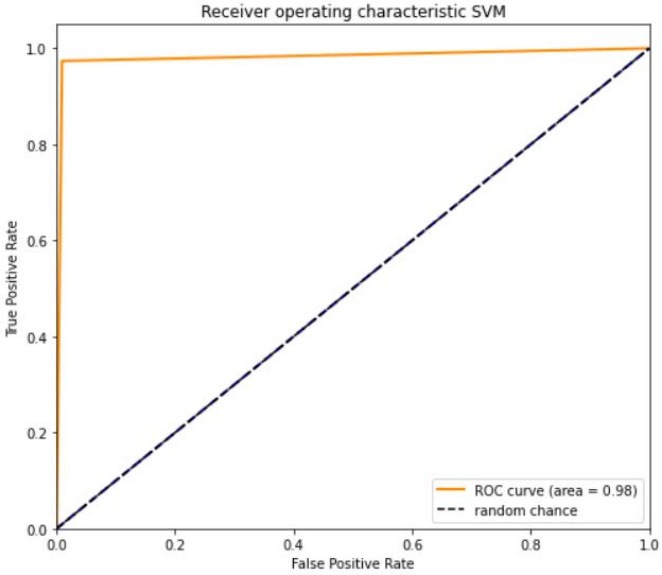

Fig. 4 ROC SVM.

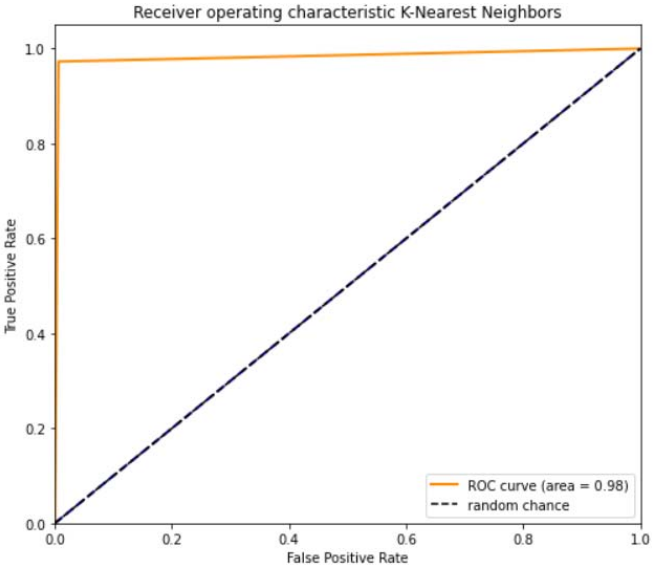

Fig. 6 ROC K-Nearest Neighbors.

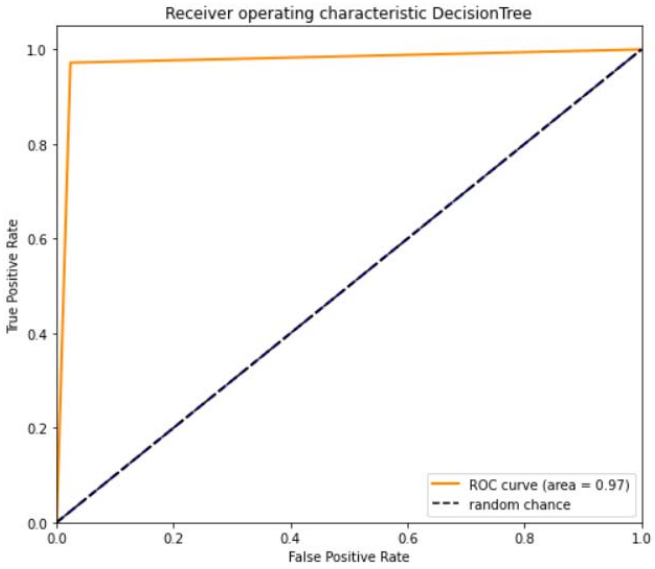

Fig. 7 ROC Decision Tree.

V. CONCLUSION

This work performs data pre-processing and uses Python with machine learning algorithms to analyze the NSL-KDD data set. We use various machine learning methods, such as decision trees, random forests, Naïve Bayes, KNN, Gradient Boosted Trees, and SVM to analyze the confusion matrix and predict the accuracy. After data pre-processing, Random Forest, KNN, and Gradient Boosted Tree can all get extremely high accuracy. The rest is good, but it is a pity that Naive Bayes did not get the desired result. This part may be because we did not perform enough pre-processing on the data to shape it into a form optimized for the model. In the future, it can be combined with software. Now it is common to compare Rapidminer and KNIME to predict whether researchers with non-program backgrounds can get higher results in cyberattacks, and they can be used quickly.

ACKNOWLEDGMENT

The authors are grateful to Ministry of Science and Technology of Taiwan for their support in terms of allocated startup research grant MOST110-2221-E-324-018.

REFERENCES

[1] R. Knight, J.R.C. Nurse, "A framework for effective corporate communication after cyber security incidents", Computers & Security, Vol. 99, id. 102036, Dec. 2020.

[2] K. Siddique, Z. Akhtar, F.A. Khan, Y. Kim, "KDD Cup 99 Data Sets: A perspective on the role of data sets in network intrusion detection research", Computer, Vol. 52, No. 2, pp. 41-51, Feb. 2019.

[3] NSL-KDD Dataset. [Online]. Available: https://www.unb.ca/cic/datasets/nsl.html, 2009.

[4] R. D. Ravipati, M. Abualkibash, "A survey on different machine learning algorithms and weak classifiers based on KDD and NSL-KDD datasets", International Journal of Artificial Intelligence and Applications, Vol. 10, No. 3, 2019.

[5] M.H. Haghighat, J. Li, "Intrusion detection system using voting-based neural network," Tsinghua Science and Technology, Vol. 26, No. 4, pp. 484-495, Aug. 2021.

[6] K. Shaukat, S. Luo, V. Varadharajan, I.A. Hameed, "A survey on machine learning techniques for cyber security in the last decade", IEEE Access, Vol. 8, pp. 222310-222354, 2020.

[7] V.A. Silva, I. Bittencourt, J.C. Maldonado, "Automatic Question Classifiers: A Systematic Review", IEEE Transactions on Learning Technologies, Vol. 12, No. 4, pp. 485-502, Oct. 2019.

[8] NSL-KDD Anomaly detection/ [Online]. Available: https://www.kaggle.com/avk256/nsl-kdd-anomaly-detection, 2020.

[9] T.T. Su, H.Z. Sun,J.Q. Zhu, S. Wang, Y.B. Li, "BAT: Deep learning methods on network intrusion detection using NSL-KDD dataset", IEEE Access, Vol. 8, pp. 29575-29585, 2020.

[10] J.F. Xu, Y.J. Zhang, D.Q. Miao, "Three-way confusion matrix for classification: A measure driven view", Information Sciences, Vol. 507, pp. 772-794, Jan. 2020.

[11] A. Tharwat, "Classification assessment methods", Applied Computing and Informatics, Vol. 17, No. 1, pp. 168-192, 2020.