Optimizing Network Anomaly Detection Based on Network Traffic

Vu Ngoc Son¹

¹Information Assurance dept. FPT University, Hanoi, Vietnam

Abstract— Cyber-attack is a very hot topic today. Nowadays, systems must always be connected to the internet, and network infrastructure keeps growing in both scale and complexity. Therefore, the problem of detecting and warning cyber-attacks is now very urgent. To improve the effectiveness of detecting cyber-attacks, many methods and techniques were applied. In this paper, we propose to apply two methods of optimizing cyber-attack detection based on the IDS 2018 dataset using Principal Component Analysis (PCA) and machine learning algorithms. In the experimental section, we compare and evaluate the efficiency of the algorithm through 2 parameters: detection and processing time, and the accuracy of the algorithm. The experimental results show that the model using optimized features has brought an apparent and better effect than models that have not reduced the feature dimension.

Keywords— PCA; Network traffic; Anomaly; cyber-attack detection.

I. INTRODUCTION

Network anomaly detection methods are usually based on two main techniques, machine learning and deep learning [1]. Deep learning algorithm often requires complex computational systems and slowly detects and alerts, so it is rarely used for network monitoring. Machine learning algorithm in this case often gives better performance because classification algorithm has a simple structure and calculates fast. Therefore, in this study, we choose to use two machine learning algorithms, namely Random forest and SVM, to detect cyber attacks [2]. In addition, in the current experimental datasets for detecting cyber-attacks, many datasets have many statistical features [1, 2]. These features could be good for other computation cases, but sometimes they are confounding factors for anomaly monitoring and detection systems using machine learning.

For that reason, in this paper, we also research and propose some methods to reduce the feature dimension in order to improve the efficiency of anomaly detection algorithms. We choose to use two dimensionality reduction methods: PCA algorithm [2, 3]; Random Search and Grid Search algorithms [4, 5].

II. RELATED WORKS

In the study [7], the authors presented the optimization method for the APT attack detection task using the UNSW-NB15 dataset, the PCA algorithm, and the correlation coefficient method. In which, the authors used the PCA algorithm to compress data and reduce feature dimensions. Besides, the correlation coefficient method assists in calculating the correlation between the features in order to remove the unimportant features. In the experimental section, we noticed that the author's evaluation method was based on the Random Forest algorithm with the change of parameters. The experimental results showed that the authors' two proposed feature dimensionality reduction brought high efficiency in both the cyber-attack detection as well as the execution time.

In the study [8], the research team proposed a method to optimize the Web-attack detection based on the profile analysis and building technique. Specifically, the research team used a technique for analyzing anomalous behaviors on each URL to analyze and detect anomalies, and then used the Elbow algorithm to automatically find the number of data groups. In the experimental section, the authors compared the proposed method with some other studies. Experimental results showed that the author's proposal brought higher efficiency.

In the study [9], Duong et al proposed a method to optimize cyber-attack detection using deep learning. In which, the author proposed to use a CNN-LSTM combined deep learning network.

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 11, November 2021)

However, the idea of using the CNN-LSTM deep learning network has been studied and presented in the study [10]. In the report [9], Duong proposed a new data pre-processing method to achieve better results.

In addition, in studies [11, 12, 13, 14, 15, 16], the authors also proposed several different cyber-attack detection approaches using supervised machine learning algorithms such as SVM, LG, Decision Tree, KNN and available datasets such as KDD 99, NSL-KDD 2000, UNSW-NB15.

III. SOME METHODS OF OPTIMIZING CYBER-ATTACK DETECTION

3.1 Introduction to the IDS2018 dataset

The CSE-CIC-IDS2018 [17] dataset includes 7 different attack scenarios: Brute-force, Heartbleed, Botnet, DoS, DDoS, Web attack, Insider attack. The attack infrastructure includes 50 devices, and the victim's organization has 5 departments consisting of 420 devices and 30 servers. The dataset includes network traffic records and system logs of each device, and 80 features extracted from network traffic collected by CICFlowMeter-V [18]. Figure 1 illustrates how to collect and process the IDS2018 dataset.

Fig. 1 The way to collect and process the IDS2018 dataset

From Figure 1, it can be seen that the IDS 2018 dataset is generated by several stages as follows:

- TCPDUMP: this is the stage of collecting data from the network flow. This data is captured by using some data collection tools.
- PCAP FILE: this is the format of the data collected by the TCPDUMP tool. Then this Pcap file is passed through two tools, Argus and Bro IDS.

- The Argus tool has the function of collecting and analyzing data. This tool analyzes Pcap data into different components according to the predefined standard and format.
- Bro-IDS tool: this is a network monitoring tool. Its function is the same as Snort or Suricata tools. After the Pcap data passes through the Bro-IDS tool, it is analyzed into fields, grouped by packet format.
- Finally, the CSV file is the result file of the process of analyzing network traffic into different features. The study [17] listed these 80 features in detail.

Table 1 below describes the dataset collected and analyzed by date.

Table 1
Statistics of the IDS2018 dataset

<i>Date of dataset</i>	<i>Attack type</i>	<i>The number of records</i>
16-02	DoS attacks - Slow HTTP Test	138,598
	Dos attacks - Hulk	41,772
21-02	DDos attacks - HOIC	205,801
	DDos attacks - LOIC - UDP	548
22-02	Brute Force - XSS	21
	Brute Force - Web	65
	SQL Injection	9
23-02	Brute Force - XSS	108
	Brute Force - Web	46
	SQL Injection	18
28-02	Infiltration	20,335
01-03	Infiltration	27,729
02-03	Bot	85,848

3.2 Optimizing cyber-attack detection

3.2.1 Data dimensionality reduction by PCA

As mentioned above, after conducting the preprocessing step, each network flow has more than 80 features. We think that need to take a step to reduce the dataset dimensionality in order to reduce the computation time as well as increase the detection ability. In this study, we propose to use the PCA dimensionality reduction method to reduce the dimensionality of the dataset.

We use the PCA method because of its effectiveness in reducing data dimensions that have been proven by many other studies. PCA doesn't remove features of the dataset like commonly used feature extraction methods. It helps to create a new feature set with a specified dimension from the old feature set while keeping the nature of the dataset. The operating principle of PCA is to try to represent the original N-dimensional dataset into another M-dimensional spatial set with $M < N$. Thus, PCA creates a new space that better represents the dataset without completely removing some features like other feature extraction methods. The study [2] also proposed the idea of using the PCA algorithm to reduce the feature dimensionality of the UNSW-NB15 dataset. The operating principle of the PCA algorithm was presented in detail in studies [2, 3].

3.2.2 Optimization using hyperparameters of the algorithm

Hyperparameter is used to control the training process so it must be set before the training process begins. Some examples of hyperparameters in deep learning are learning rate and batch size. If a hyperparameter is too small or too big corresponding to the model power being too large or too small, the test error will be high. Two methods of automatically fine-tuning hyperparameters are Random Search and Grid Search [2, 4, 5, 6].

- *Random Search:* this method only randomly selects a finite number of elements from the above list to test the model. In reality, Random Search randomly selects parameter pairs to test the model's accuracy. The study [19] presented this algorithm in detail.
- *Grid Search:* Grid search is a technique for fine-tuning hyperparameters. It could facilitate model building and model evaluation for any combination of algorithmic hyperparameters per grid. In Grid Search, all combinations of hyperparameters are passed one by one into the model and tested on each model. It gives us the hyperparameter set for the best score. The cross-validation in the Scikit-learning package is used to automatically iterate over these hyperparameters. The steps to apply the Grid Search model are as follows: splitting the dataset into training/testing dataset; using GridSearch with 5-Fold Cross-validation to train and test models (RF and SVM) to get the best model with an optimal combination of hyperparameters; then calculating metrics (Accuracy, Recall, F1-score) on each model.

Finally, using the model to predict the classification results, and comparing them with actual labels.

IV. SOME METHODS OF OPTIMIZING CYBER-ATTACK DETECTION

4.1 Experimental scenario

In this paper, we use two basic algorithms, namely RF and SVM, for comparison. Next, we proceed to optimize the features of the IDS2018 dataset by two methods: PCA and hyperparameters of the algorithm. After obtaining this information, we proceed to compare the experimental results with each other.

4.2 Experiments and evaluations of anomaly detection by feature dimensionality reduction

4.2.1 Feature dimensionality reduction results

In this step, we use PCA to convert the old dataset into a new dataset with smaller dimensions to help reduce the computation time in the classification. To find the best number of dimensions to reduce, we plotted and found that the model with the parameter `n_components=16` gave the best results. The graph is shown in the figures below.

Fig. 2 Correlation graph between variance and dimension of the dataset

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 11, November 2021)

Thus, after using the PCA algorithm, the feature dimension is 16 instead of 80 as the original. Next, we compare the results before and after reducing the feature dimension of the dataset.

4.2.2 Evaluating the feature dimensionality reduction model

Tables 2 and 3 below show the experimental results of anomaly detection based on RF and SVM algorithms using datasets before and after feature dimensionality reduction.

Table 2:
Comparison of the detection ability of the new dataset with the RF algorithm

Runs	Dataset		RF with original dataset and default hyperparameters			RF after performing PCA and default hyperparameters		
	Train	Test	Accuracy	F1-score	Time	Accuracy	F1-score	Time
1	209715	104858	83.229%	80.350%	0.1173	67.832%	68.118%	0.1236
2	209715	104858	34.524%	51.217%	0.1130	99.918%	99.881%	0.1238
3	209715	104858	34.596%	51.293%	0.1149	85.586%	82.669%	0.1276
4	209715	104858	35.711%	51.720%	0.1171	70.900%	70.286%	0.1243
5	209715	104858	95.474%	93.822%	0.1182	99.193%	98.840%	0.1231

Table 3:
Comparison of the detection ability of the new dataset with the SVM algorithm

Runs	Dataset		SVM with original dataset and default hyperparameters			SVM after performing PCA and default hyperparameters		
	Train	Test	Accuracy	F1-score	Time	Accuracy	F1-score	Time
1	209715	104858	99.992%	99.989%	4.3343	99.825%	99.745%	1.3482
2	209715	104858	99.992%	99.989%	3.9310	99.823%	99.743%	1.1689
3	209715	104858	99.928%	99.895%	4.4616	99.820%	99.739%	1.4174
4	209715	104858	99.992%	99.989%	4.1549	99.806%	99.720%	1.2987
5	209715	104858	99.739%	99.621%	4.3532	99.921%	99.885%	1.3539

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 11, November 2021)

Through test results of classification by two machine learning algorithms with the new dataset created by PCA in tables 2 and 3, it can be seen that:

Regarding the RF algorithm, according to Table 2, seeing that the accuracy results of classification with the new dataset are markedly improved compared to the original dataset with full features. The reason is that by using the PCA method to reduce the data dimension to 16, we have obtained a new dataset represented in a completely new space that considerably improves the ability to correctly detect.

Regarding the SVM algorithm, according to the results summarized in Table 3, with the new dataset consisting of 16 features, the SVM algorithm has the same ability of accurately detecting as using the original dataset with 78 features. Moreover, the classification time of the SVM algorithm with the new dataset is markedly reduced (about more than 3 times) compared to using the original dataset. The reason is that with the dataset with a smaller spatial, the SVM algorithm just needs to calculate the smaller number of distances to return a hyperplane for the detection process.

With experimental results using RF and SVM algorithms, it can be seen that the new dataset generated by using the PCA method had the same or better classification ability than the original dataset with full features. Besides, the classifying speed is also markedly improved due to the smaller size of the new dataset.

4.3 Experiment when using hyperparameters

4.3.1 Finding hyperparameters

The hyperparameter dictionary of machine learning algorithms is listed in the tables below.

Table 4:
The hyperparameter dictionary of the RF algorithm

```
n_estimators = [5,10,20,40,80,100]

max_features = ['auto', 'sqrt', 'log2']

max_depth = [int(x) for x in np.linspace (start=1,stop= 16,
num = 16)]
```

Table 5:
The hyperparameter dictionary of the SVM algorithm

```
kernel = ['linear', 'poly', 'rbf', 'sigmoid']

gamma = ['scale', 'auto']
```

After running the two algorithms Grid Search and Random Search, we got the hyperparameter sets with the highest scores as follows:

Random Forest algorithm

RandomizedSearchCV runs 300 fits

Best Parameters RandomizedSearchCV: {'n_estimators': 10, 'max_features': 'log2', 'max_depth': 7}

GridSearchCV runs 864 fits

Best Parameters GridSearchCV: {'max_depth': 5, 'max_features': 'auto', 'n_estimators': 80}

SVM algorithm

RandomizedSearchCV runs 18 fits

Best Parameters RandomizedSearchCV: {'kernel': 'poly', 'gamma': 'auto'}

GridSearchCV runs 24 fits

Best Parameters GridSearchCV: {'gamma': 'auto', 'kernel': 'poly'}

Grid Search and Random Search results in SVM algorithm are the same.

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 11, November 2021)

4.3.2 Experiments and evaluating hyperparameters

Table 6:

Experimental results with two hyperparameter sets obtained after using two methods, Grid Search and Random Search, with RF algorithm

<i>RF after performing PCA and default hyperparameters</i>			<i>RF after performing PCA and Random Search hyperparameters</i>			<i>RF after performing PCA and Grid Search hyperparameters</i>		
<i>Accuracy</i>	<i>F1-score</i>	<i>Time</i>	<i>Accuracy</i>	<i>F1-score</i>	<i>Time</i>	<i>Accuracy</i>	<i>F1-score</i>	<i>Time</i>
67.832%	68.118%	0.1236	67.307%	67.765%	0.0188	67.307%	68.112%	0.0986
99.918%	99.881%	0.1238	90.353%	68.204%	0.0600	90.681%	87.983%	0.0950
85.586%	82.669%	0.1276	74.809%	71.333%	0.0196	84.416%	79.778%	0.1009
70.900%	70.286%	0.1243	42.467%	54.484%	0.0186	86.138%	83.240%	0.1003
99.193%	98.840%	0.1231	97.081%	95.925%	0.0195	99.780%	99.680%	0.0986

Table 7:

Experimental results with two hyperparameter sets obtained after using two methods, Grid Search and Random Search, with SVM algorithm

<i>SVM after performing PCA and default hyperparameters</i>			<i>SVM after performing PCA and Random Search/ Grid Search hyperparameters</i>		
<i>Accuracy</i>	<i>F1-score</i>	<i>Time</i>	<i>Accuracy</i>	<i>F1-score</i>	<i>Time</i>
99.825%	99.745%	1.3482	99.630%	99.461%	0.0977
99.823%	99.743%	1.1689	99.291%	98.974%	0.1176
99.820%	99.739%	1.4174	98.887%	98.407%	0.0848
99.806%	99.720%	1.2987	98.279%	97.559%	0.1179
99.921%	99.885%	1.3539	99.509%	99.300%	0.0978

After evaluating the newly generated dataset by the PCA method, seeing that the new dataset has better detection ability than the full original dataset. However, the hyperparameters of the algorithms are being set as default. Therefore, we need to find a better hyperparameter set for each algorithm. With this requirement, we have performed to find the most optimal sets of hyperparameters for two popular machine learning algorithms, RF and SVM, by two methods: Grid Search and Random Search.

With the RF algorithm, Grid Search and Random Search algorithms gave two different results. Next, we put two hyperparameter sets into the model to classify on the CIC-2018 dataset for evaluation. The results of the process of evaluating the classification ability are presented in Tables 6 and 7.

With the RF algorithm, the results show that the hyperparameter set selected by Grid Search had the same or higher accuracy than using the default hyperparameter set. Moreover, the detection time of the proposed method with the RF algorithm is markedly reduced compared to using the original hyperparameter set. However, with the hyperparameter set found by Random Search, the classification results are lower than that of the default hyperparameter set. The reason is that Random Search only randomly selects some combinations of hyperparameters for evaluation. Therefore, the final hyperparameter set may not be the best one. Specifically, in this case, the set of hyperparameters found by Random Search had lower results than that of the default hyperparameter set.

With the SVM algorithm, the Grid Search and Random Search methods yielded the same hyperparameter sets. Then, we use two hyperparameter sets for classifying on the CIC-2018 dataset for evaluation. The results show that the accuracy when using the selected hyperparameter set is nearly equivalent to that of the default hyperparameter set. Regarding the detection time of the proposed method with the SVM algorithm, the detection time of the proposed method is reduced from 10 to 17 times compared to using the SVM algorithm with a default hyperparameter set.

The above results show that the proposed method could markedly improve the performance and detection speed of machine learning algorithms. This helps save system power and improve the ability to detect anomalies in real-time.

V. CONCLUSION

In this paper, we have presented methods of finding a better feature set by the PCA method and optimizing hyperparameters of some machine learning algorithms for detecting network anomalies. Experimental results with the PCA method show that with the used dataset, the most optimal dimension after reduction is 16. Next, we have found the best hyperparameter sets using Grid Search and Random Search methods for two machine learning algorithms, RF and SVM. We then have put these hyperparameter sets into the machine learning model to classify. The results show that the proposed method has improved markedly the classification time of the machine learning algorithm with the same or higher accuracy than using conventional methods. From the achieved results, we noticed that the proposed method is able to apply in optimizing the dataset as well as the hyperparameter sets for detecting abnormal behaviors on the internet effectively.

REFERENCES

- [1] Gilberto Fernandes Jr., Joel J. P. C. Rodrigues, Luiz Fernando Carvalho, Jalal F. Al-Muhtadi & Mario Lemes Proença Jr.: A comprehensive survey on network anomaly detection. *Telecommunication Systems*, vol. 70, pp. 447–489 (2019)
- [2] James Bergstra, Yoshua Bengio. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research* 13 (2012) 281-305
- [3] J. Shlens.: A Tutorial on Principal Component Analysis. (2014) arxiv.org/, arXiv:1404.1100
- [4] Zeldia B. Zabinsky: Random Search Algorithms. <https://doi.org/10.1002/9780470400531.eorms0704>.
- [5] Jean-Marie Dufour, Julien Neves: Chapter 1 - Finite-sample inference and nonstandard asymptotics with Monte Carlo tests and R. *Handbook of Statistics*, vol. 41, pp. 3-31 (2019)
- [6] Brownlee: Hyperparameter Optimization With Random Search and Grid Search. *Machine Learning Mastery* (2020). <https://machinelearningmastery.com/hyperparameter-optimization-with-random-search-and-grid-search/> (accessed Jun. 02, 2021)
- [7] Cho Do Xuan, Nam Nguyen, Hoa Nguyen Dinh. An adaptive anomaly request detection framework based on dynamic web application profiles. *International Journal of Electrical and Computer Engineering (IJECE)* Vol.10, No.5, pp. 5335~5346
- [8] Cho Do Xuan, Hoang Thanh, Nguyen Tung Lam: Optimization of network traffic anomaly detection using machine learning. *International Journal of Electrical and Computer Engineering*, vol. 11(3), pp. 2360-2370 (2021).

International Journal of Emerging Technology and Advanced Engineering

Website: www.ijetae.com (E-ISSN 2250-2459, Scopus Indexed, ISO 9001:2008 Certified Journal, Volume 11, Issue 11, November 2021)

- [9] Lai Van Duong. Optimization of Cyber-Attack Detection Using the Deep Learning Network. International Journal of Computer Science and Network Security, VOL.21 No.7. 2021, 159- 169.
- [10] Do Xuan, C., Dao, M.H. A novel approach for APT attack detection based on combined deep learning model. Neural Comput & Applic 33, 13251–13264 (2021). <https://doi.org/10.1007/s00521-021-05952-5>
- [11] Li Y., Ma R. and Jiao R.: A Hybrid Malicious Code Detection Method Based on Deep Learning. International Journal of Security and Its Applications (IJSIA), vol. 9(5), pp. 205–216 (2015).
- [12] K. Vikash., et al.: An integrated rule based intrusion detection system: analysis on UNSW-NB15 data set and the real time online dataset. Cluster Computing, vol. 22 (2019) doi: 10.1007/s10586-019-03008-x
- [13] N. Moustafa., et al.: Novel Geometric Area Analysis Technique for Anomaly Detection using Trapezoidal Area Estimation on Large-scale Networks. IEEE Transactions on Big Data, vol. 5(4), pp. 2332-7790 (2017)
- [14] N. Moustafa et al.: Big Data Analytics for Intrusion Detection System: Statistical Decision-Making Using Finite Dirichlet Mixture Models. (2017) doi :10.1007/978-3-319-59439-2_5
- [15] S. Bagui, et al.: Using machine learning techniques to identify rare cyber-attacks on the UNSW-NB15 dataset. Security and Privacy (2019) doi: 10.1002/spy2.91
- [16] S. Rajagopal, et al.: Performance analysis of binary and multiclass models using azure machine learning. International Journal of Electrical and Computer Engineering (IJECE), vol. 10(1), pp. 978-986.
- [17] CSE-CIC-IDS2018 on AWS. <https://www.unb.ca/cic/datasets/ids-2018.html>.
- [18] CICFlowMeter (formerly ISCXFlowMeter). <https://www.unb.ca/cic/research/applications.html#CICFlowMeter>