ISSN 2347 - 3983
Volume 9. No. 7, July 2021
International Journal of Emerging Trends in Engineering Research
Available Online at http://www.warse.org/IJETER/static/pdf/file/ijeter20972021.pdf
https://doi.org/10.30534/ijeter/2021/20972021

VPN Encrypted Traffic classification using XGBoost

Sami Smadi[a], Omar Almomani[b], Adel Mohammad[c], Mohammad Alauthman[d], Adeeb Saaidah[e]
[abe]Department of Information System and Networks, Faculty of Information Technology, The World Islamic Science and Education University, Amman, Jordan
[c]Department of Computer Science, Faculty of Information Technology, The World Islamic Science and Education University, Amman, Jordan
[d]Department of Information Security, Faculty of Information Technology, University of Petra, Amman, Jordan
Email:{sami.smadi[a], omar.almomani[b], Adel Mohammad[c], adeeb.saaidah[e]}@wise.edu.jo, mohammad.alauthman[d]@uop.edu.jo

ABSTRACT

Classification network traffic are becoming ever more relevant in understanding and addressing security issues in Internet applications. Virtual Private Networks (VPNs) have become one famous communication forms on the Internet. In this study, a new model for traffic classification into VPN or non-VPN is proposed. XGBoost algorithm is used to rank features and to build the classification model. The proposed model overwhelmed other classification algorithms. The proposed model achieved 91.6% accuracy which is the highest registered accuracy for the selected dataset. To illustrate the merit of the proposed model, a comparison was made with sixteen different classification algorithms.

Key words: VPN, XGBoost, Encrypted traffic, ensemble learning, Network traffic classification.

1. INTRODUCTION

Traffic Classification is the principle of recognition of protocols and implementations by evaluating the network traffic. Traffic classification techniques are used for a wide variety of purposes, including Quality of Service (QoS), traffic forming, Intrusion Detection Systems (IDS), and network forensic solutions [1, 2, 3, 4, 5, 6]. Usually, traffic can be classified as normal or malware traffic to detect and prevent attacks. Traffic encryption has increasingly become common for the widespread use of encryption methods in network applications [7, 8, 9]. Many malware use encryption methods like TLS to encrypt information circulation to prevent detection in the intrusion detection mechanism implemented in firewalls and IDS. Therefore, the conventional methods of classification of traffic are facing new challenges [10].

Traffic classification can be classified according to its final purposes into three groups. Firstly, encrypted traffic, Secondly, encapsulation of protocols (e.g. tunneled by VPN or HTTPS). Finally, according to particular applications (e.g. Skype) or device form (e.g. Downloading, Chat) [10]. Some program supports many services, such as chat, VOIP, file transfer, etc., such as Skype or Facebook.

VPN is becoming a common means of concealing hackers' online activities [11]. This is supported by VPNs' easy use, which is no longer just a remote tool for connections to company services. Suppose attackers choose to remotely access the company's network to capture business secrets. In that case, they can make a VPN (or multiple VPNs) look as if they were legitimate users infiltrating their network, or to conceal their locations. The offenders who carried out such assaults could prove difficult if not impossible if they used VPNs to hide their identity. Therefore, deciding if a VPN has been used or not may help to trace the perpetrators of the attacks mentioned above. In this article, we classify encrypted traffic and encrypted traffic tunneled by VPN. The classification of VPN traffic remains an issue to be addressed. VPN tunnels protect the anonymity of shared data over the physical network link, including packet-level encryption, making it very difficult to detect programs that run via the VPN services.

There are four major traffic classification methods: ports-based, Deep Packet Inspection (DPI) based, mathematical-based, and behavioral-based. The port-based process's exactness is very poor since random port, and port disguise is widely applied. The DPIbased approach has great difficulties, as the encrypted traffic cannot be decrypted. The ongoing inquiry focuses largely on mathematical and behavioral methods [12, 13]. Machine learning techniques can solve some drawbacks in port and payload approaches. More precisely, the Internet traffic can be categorized by using separate traffic statistics from the application protocol, such as flow time, packet length variances, maximum or minimum segment size, window size, round trip time, and packet time inter-arrival.

Our contribution in this paper has two folds. Firstly, a new classification model is proposed,

which classifies network traffic into VPN and non-VPN traffic using the GXBoost algorithm. Secondly, the computational overhead is reduced by reducing the set of features to a set that can be extracted with low computational complexity. Moreover, the list of features was sorted based on the most imported ones in the classification process. The remainder of this paper is organized as follows: Section 2 presents the related work. In section 3 Authors describe the proposed model. The findings obtained are summarized and discussed in Section 5. Lastly, the conclusion and future studies are discussed in section 6.

2. **RELATED WORK**

Researches that classify traffic based on packet size and flow-based features began in the early ninety's [14, 15]. These studies prove that statistical attributes such as packet length, interarrival times, and flow duration can be used to track protocols. A QoS classifier was proposed by Caicedo-Mu˜noz et al. [16], the proposed model classifies VPN traffic for a particular domain based on per-hop activity (PHB). Time-related features were discovered, especially for VPN traffic. A baseline QoS-Marked dataset was generated from a characterized VPN traffic; different machine learning algorithms (MLA) were compared, and a T-Tester was performed. Based on the obtained results, the learning model has the best behavior for all scenarios with 94,42% accuracy. Miller et al. proposed a multi-layered perceptron neural network model to classify network traffic into VPN or non-VPN [11]. The proposed model depends on TCP flow-based features to classify network traffic as either VPN or not VPN. The flow-based features are founded based on Pearson's Correlation Coefficient model. The accuracy of the proposed model was 92%. The reliability of time-related features has been studied to solve the difficult issue of encrypted traffic characterization and VPN traffic detection by Gil et al [17]. The features related to the time model were proposed, and the classification algorithms used were C4.5 and KNN. The technique proposed shows that time-based features are useful in detecting VPN transmission and reaching accuracy levels above 80%. In all experiments, C4.5 and KNN performed similarly, although C4.5 managed to achieve an improved outcome. The study has also found that C4.5 and KNN perform better when the flows are generated using shorter timeout values (15 seconds), which contradicts the common assumption of using 600s as timeout duration. Traffic classification SVM model based on radial base kernel function has been proposed by Z Fan and R Liu [12]. After feature optimization, thirteen features were selected. Overall classification accuracy of 98% among all the traffic classes is achieved. The main drawback of the proposed model is that it did not consider encrypted traffic such as VPN, which is becoming a popular way to mask the online activities of attackers. In the literature, a variety of system classification approaches were proposed to classify traffic correctly based on flow and packet-based characteristics. The traffic classification for encapsulated traffic is difficult and thus not widely explores in the literature. In this paper, we focus on traffic classification into VPN and non-VPN traffic using XGBoost.

3. **PROPOSED MODEL**

3.1. Dataset Description

Gil et al. have created a representative dataset that captures real traffic that contains regular traffic and VPN traffic [17]. The dataset includes seven different traffic types captured from protocols and applications. The captured traffic contains web browsing, email, chat, streaming, file transfer, VoIP, and P2P. For each traffic type, there are two versions like VOIP and VPN-VOIP. The selected dataset contains 23 features which are fully described by Gil et al. [17]. In this study, the dataset with a fifteen-second flow timeout value is considered since it has been proved by Gil et al. and Lashkari, A.H., et al. [17, 18] that it produces the best results in terms of precision and recall.

3.2. System Model Based on XGBoost

The proposed model to classify the encrypted traffic to VPN or Non-VPN is shown in Figure1, which is described as follows: Firstly, the dataset collected in [17] is divided into training and testing dataset, where training dataset is used to build the trained model and the testing dataset used to test the resulting model. Secondly, XGBoost is used to select and rank the most important feature in the classification process.

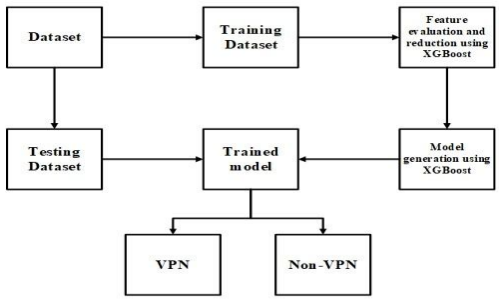

Figure1: System Model.

Thirdly, the model is generated by creating a set of decision trees and combine them in one decision tree that more accurate than the previous one as shown in Figure2. Finally, the generated model is tested, and the performance of the register model is calculated.

XGBoost stands for "Extreme Gradient Boosting" which is a common and effective open-source algorithmfor gradient boosted trees [19]. Gradient boosting is a supervised learning algorithm that seeks to predict an objective variable accurately by integrating the predictions of several weaker models [20]. XGBoost is another tree model, a common data mining tool with high speed and performance. The XGBoost model will compute 10 times as quickly as the Random Forest. The XGBoost model created using the additive tree method, where a new tree is added in each stage to complement the trees already constructed. This generally enhances the accuracy as more trees are constructed. The final answer is the weighted sum of each tree's predictions, or the optimal linear combination of every decision-making body can be told as shown in Figure2.

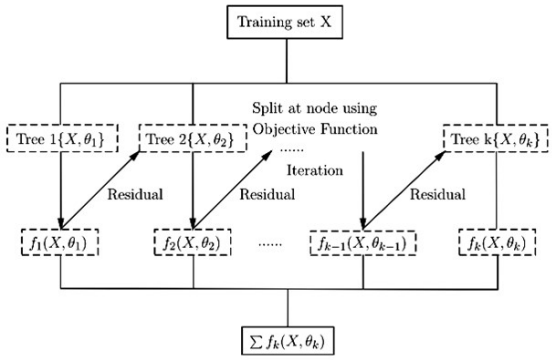

Figure 2. XGBoost Model generation [21].

Based on XGBoost Algorithm described by P. Deven and N. Khare [22], the decision tree is constructed.
Where *x*1, *x*2,..., x_m is the set of features. Each feature's rank is computed based on the number of times that feature is used to split the training data in each version of the decision tree used in building the final model.

3.3. Evaluation Metrics

To validate and analyze the proposed model, we use a set of normative measurements and indices, such as accuracy, precision, the area under the ROC curve, and detection rates metrics. The efficiency of the classifier has been calculated according to the confusion matrix (CM) as shown in Figure3. For each record in the testing dataset the following evaluation metric is applied:

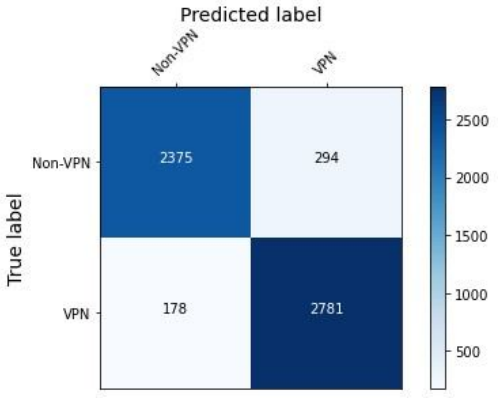

Figure 3. Confusion Matrix.

- True-positive CM[1][1]: the number of VPN packets that are correctly classified as VPN.
- True-negative CM[0][0]: the number of non-VPN packets that are correctly classified as non-VPN.
- False-Positive CM[0][1]: the number of non-VPN packets that are incorrectly classified as VPN.
- False-negative CM[1][0]: the number of VPN packets incorrectly classified as non-VPN.

Based on the previous evaluations, the following metrics can be derived:

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

$$Recall = \frac{TP}{(TP+FN)} \quad (2)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (3)$$

- The Area Under Curve (AUC) [23]: ROC is a curve of probability, and AUC is the separability indicator. It demonstrates the possibility of differentiating classes in the proposed model.
- Mean Square Error (MSE)[24]:

$$MSE = \frac{\sum_{i=1}^{n}(o_i - T_i)^2}{n} \quad (4)$$

Where o_i is the model predictions for the *ith* packet,$i_i$is the desired target for the same packet, *n* is the number of the packet. The parameters value that used to tune the XGBoost is presented in table1

Table1: XGBoost Key parameters

Parameter	**Default Value**
maximum depth (max depth)	5
learning rate (eta)	0.3
Numberofgeneratedtrees(nrounds)	100–1000
minimum loss reduction for splitting node (gamma)	0
minimum sum of samples weight of all the observations required in a child (min child weight)	1
# of features that used to find the best node split (colsample bytree)	1
Objective function	binary:logistic

4. **RESULT AND DISCUSSION**

Table 2 shows the result of the proposed model in comparison with the most famous machine learning algorithms[26]. The result of the proposed model was validated using 10-fold cross-validation. The proposed model to classify traffic to VPN or non-VPN shows a promising result, where it registered the highest results compared to other classifiers in terms of accuracy, MSE, precision, recall, and AUC. Sixteen classifiers were selected for the comparison the Bagging, RandomForest, and DecisionTree classifiers show the best accuracy with 90.5%, 89.3%, and 88.3% respectively. Figure4 shows the accuracy of the proposed model in comparison with a set of machine learning algorithms, and the highest accuracy register is for the proposed model with 91.6 %.

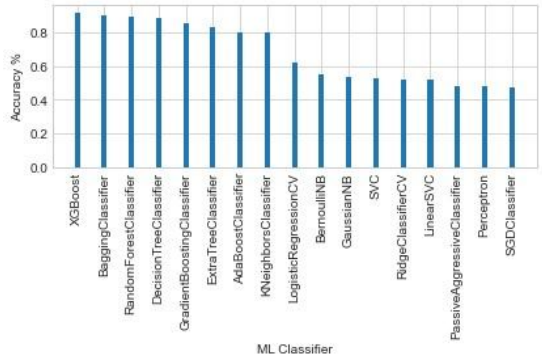

Figure 4: Compare the proposed model accuracy with other MLA.

Figure5 shows the Log Loss [25] of the proposed model, where the x-axis represents the number of epochs, and the y-axis represent the log loss. Figure6 shows the classification errors for the proposed model. To measure the predictions' performance, we shall use the Logloss [25] indicator. The Log Loss function is commonly used for evaluation.

$$LogLoss = \frac{1}{N}\sum_{i=1}^{N}\sum_{j=1}^{M}\sum y_{i,j}\log\left(x_{i,j}\right) (4)$$

Where N represents is the total samples of the dataset.

M is the number of class labels is 1 when the observation is in class, otherwise the value of is 0. indicates the predicted probability of sample is in class. The logloss rate in train and test data set for XGBoost logs can be seen in Figure5. The XGBoost's Logloss test function was successful with a test logLoss of 0.2 percent when compared to the training set. This is good for testing data and is very likely to improve by training on more data with the XGBoost algorithm.

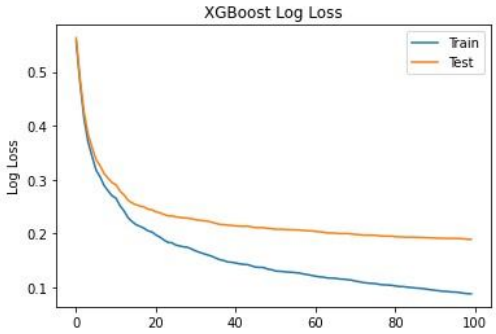

Figure 5: XGBoost Log Loss.

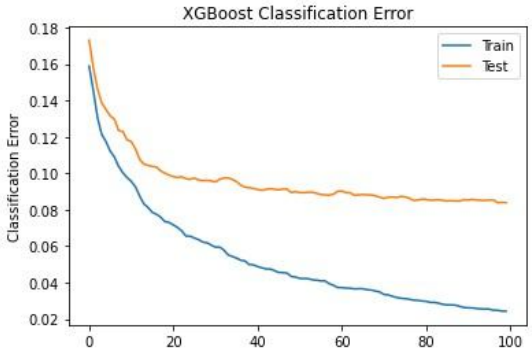

Figure 6. XGBoost Classification Error.

Table2 :Comparison of the proposed model with another machine learning algorithm.

Algo. No.	MLA Name	Accuracy	MSE	Precision	Recall	AUC
1	Proposed model (XGBoost)	0.9160	0.0830	0.9040	0.9400	0.9800
2	BaggingClassifier	0.9054	0.094616	0.903299	0.916688	0.904877
3	RandomForestClassifier	0.8935	0.106477	0.891403	0.90621	0.892955
4	DecisionTreeClassifier	0.8831	0.116871	0.87931	0.89931	0.882404
5	GradientBoostingClassifier	0.8554	0.14459	0.823866	0.919244	0.852549
6	ExtraTreeClassifier	0.8346	0.165378	0.836016	0.849476	0.833956
7	AdaBoostClassifier	0.8021	0.197894	0.774785	0.874776	0.798847
8	KNeighborsClassifier	0.8002	0.19976	0.8163	0.796064	0.800427
9	LogisticRegressionCV	0.6195	0.380464	0.615755	0.719141	0.615071
10	BernoulliNB	0.5506	0.44936	0.586894	0.466905	0.554394
11	GaussianNB	0.5324	0.467617	0.528244	0.9655	0.512964
12	SVC	0.5264	0.473614	0.524071	0.998722	0.505209
13	RidgeClassifierCV	0.5235	0.476546	0.58043	0.310759	0.53299
14	LinearSVC	0.5217	0.478278	0.580758	0.297726	0.531764
15	PassiveAggressiveClassifier	0.4836	0.516391	0.503997	0.612318	0.477838
16	Perceptron	0.4832	0.516791	0.579909	0.032456	0.503418
17	SGDClassifier	0.4765	0.523454	0.498011	0.479939	0.476394

Table 3 shows the list of available features in the dataset with their rank sorted from the highest to the lowest important. These values were calculated using the XGBoost algorithm as stated in Figure1. To evaluate the proposed model and select the best number of features the model accuracy was calculated as shown in table4. At the beginning the accuracy of the proposed model for all features is

Table3: Feature ranking using XGBoost Algorithm.

Feature ID	Feature Name	Rank	Feature ID	Feature Name	Rank
1	max_flowiat	0.164207	13	mean_idle	0.031719
2	total_biat	0.095933	14	min_fiat	0.029812
3	min_idle	0.069941	15	min_biat	0.028861
4	min_flowiat	0.063247	16	mean_fiat	0.028502
5	mean_flowiat	0.057268	17	duration	0.027158
6	max_fiat	0.053839	18	min_active	0.02463
7	std_flowiat	0.050517	19	std_idle	0.019076
8	max_biat	0.046376	20	mean_active	0.01774
9	flowPktsPerSecond	0.044891	21	max_active	0.016427
10	mean_biat	0.043076	22	std_active	0.011798
11	total_fiat	0.04126	23	max_idle	0
12	flowBytesPerSecond	0.033721			

calculated, then the experiment is repeated, and the following steps are applied. Firstly, the feature with the lowest rank is deleted from the dataset. Secondly, rebuild the model using the XGBoost algorithm for the new dataset. Lastly, test and evaluate the model. In the second iteration, the second lowest feature is deleted and apply the previous steps and so on.

Table 4: Accuracy of the proposed model after feature reductions.

No. Feature	Accuracy	No. Feature	Accuracy
23	0.916134	11	0.89801
22	0.916134	10	0.891791
21	0.917377	9	0.890014
20	0.918977	8	0.886461
19	0.918443	7	0.887704
18	0.915601	6	0.883618
17	0.918266	5	0.870469
16	0.915245	4	0.846304
15	0.915601	3	0.817164
14	0.913113	2	0.811656
13	0.913646	1	0.702026
12	0.914712		

Table 4 indicates that the max flowiat is the most important feature, and it alone can be used to predict the VPN traffic with 70.20 % accuracy. Moreover, the first 12 features in table3, can produce the same accuracy level as all features that can be used to

enhance the detection mechanism and speed up feature extraction.

5. **CONCLUSION**

In this paper, a new model is proposed for encrypted traffic classification into VPN and non-VPN. Seventeen different classifiers were compared to determine the best one in classifying encrypted traffic. XGBoost shows the best result with 91.6 % accuracy, and these results were validated using 10-fold cross-validation. The proposed model also shows the same level of accuracy after reducing the number of features to twelve features, which have a high impact on feature extraction from the Realtime traffic.

REFERENCES

[1] T. S. Tabatabaei, M. Adel, F. Karray, M. Kamel, Machine learning-based classification of encrypted internet traffic, in: International Workshop on Machine Learning and Data Mining in Pattern Recognition, Springer, 2012, pp. 578–592. doi:https: //doi.org/10.1007/978-3-642-31537-4_45.

[2] D. Kshirsagar, S. Kumar, An efficient feature reduction method for the detection of dos attack, ICT Expressdoi:https:// doi.org/10.1016/j.icte.2020.12.006.

[3] V. Kanimozhi, T. P. Jacob, Artificial intelligence based network intrusion detection with hyper-parameter optimization tuning on the realistic cyber dataset cse-cic-ids2018 using cloud computing, ICT Express 5 (3) (2019) 211–214. doi:https: //doi.org/10.1016/j.icte.2019.03.003.

[4] O. Almomani, A feature selection model for network intrusion detection system based on pso, gwo, ffa and ga algorithms, Symmetry 12 (6) (2020) 1046.

[5] M. Madi, F. Jarghon, Y. Fazea, O. Almomani, A. Saaidah, Comparative analysis of classification techniques for network fault management, Turkish Journal of Electrical Engineering & Computer Sciences 28 (3) (2020) 1442–1457.

[6] O. Almomani, A hybrid model using bio-inspired metaheuristic algorithms for network intrusion detection system, CMC-COMPUTERS MATERIALS & CONTINUA 68 (1) (2021) 409–429.

[7] Z. Cao, G. Xiong, Y. Zhao, Z. Li, L. Guo, A survey on encrypted traffic classification, in: International Conference on Applications and Techniques in Information Security, Springer, 2014, pp. 73–81. doi:https://doi.org/10. 1007/978-3-662-45670-5_8.

[8] M. Alauthman, Anefficient approach to online bot detection based on a reinforcement learning technique, Ph.D. thesis, Northumbria University (2016).URLhttp://nrl.northumbria.ac.uk/id/eprint/ 29617/

[9] M. ALAUTHMAN, P2P bot detection using deep learning with traffic reduction schema, Journal of Theoretical and Applied Information Technology 98 (15).

[10] W. Wang, M. Zhu, J. Wang, X. Zeng, Z. Yang, End-to-end encrypted traffic classification with one-dimensional convolution neural networks, in: 2017 IEEE International Conference on Intelligence and Security Informatics (ISI), IEEE,2017,pp.43–48.doi:https://doi.org/10.1109/ISI.2017.8004872

[11] S. Miller, K. Curran, T. Lunney, Multilayer perceptron neural network for detection of encrypted vpn network traffic, in: 2018 International Conference On Cyber Situational Awareness, Data Analytics And Assessment (Cyber SA), IEEE, 2018, pp. 1–8. doi:https://doi.org/10.1109/CyberSA.2018. 8551395.

[12] Z. Fan, R. Liu, Investigation of machine learning based network traffic classification, in: 2017 International Symposium on Wireless Communication Systems (ISWCS), IEEE, 2017, pp. 1–6. doi:https://doi.org/10.1109/ISWCS.2017. 8108090.

[13] M. Alauthman, N. Aslam, M. Al-Kasassbeh, S. Khan, A. AlQerem, K.-K. R. Choo, An efficient reinforcement learningbased botnet detection approach, Journal of Network and Computer Applications 150 (2020) 102479. doi:https://doi. org/10.1016/j.jnca.2019.102479.

[14] V. Paxson, Empirically derived analytic models of wideareatcp connections, IEEE/ACM transactions on Networking 2 (4) (1994) 316–336. doi:https://doi.org/10.1109/90.330413.

[15] V. Paxson, S. Floyd, Wide area traffic: the failure of poisson modeling, IEEE/ACM Transactions on networking 3 (3) (1995) 226–244. doi:https://doi.org/10.1109/90.392383.

[16] J. A. Caicedo-Munoz, A. L. Espino, J. C. Corrales, A. Rend˜ on,´Qos-classifier for vpn and non-vpn traffic based on time-related features, Computer Networks 144 (2018) 271–279. doi: https://doi.org/10.1016/j.comnet.2018.08.008.

[17] G. Draper-Gil, A. H. Lashkari, M. S. I. Mamun, A. A. Ghorbani, Characterization of encrypted and vpn traffic using time-related, in: Proceedings of the 2nd international conference on information systems security and privacy (ICISSP), 2016, pp. 407–414. doi:10.5220/0005740704070414.

[18] A. H. Lashkari, G. Draper-Gil, M. S. I. Mamun, A. A. Ghorbani, Characterization of tor traffic using time based features., in: ICISSp, 2017, pp. 253–262. doi:10.5220/0006105602530262.

[19] J. H. Friedman, Greedy function approximation: a gradient boosting machine, Annals of statistics (2001) 1189–1232. URL http://www.jstor.org/stable/2699986

[20] T. Chen, C. Guestrin, Xgboost: A scalable tree boosting system, in: Proceedings of the 22nd acmsigkdd international conference on knowledge discovery and data mining, 2016, pp. 785–794. doi:https://doi.org/10.1145/2939672.2939785.

[21] D. Zhang, L. Qian, B. Mao, C. Huang, B. Huang, Y. Si, A datadriven design for fault detection of wind turbines using random forests and xgboost, IEEE Access 6 (2018) 21020–21031. doi:10.1109/ACCESS.2018.2818678.

[22] P. Devan, N. Khare, An efficient xgboost–dnn-based classification model for network intrusion detection system, Neural Computing and Applications (2020) 1–16doi:https://doi.org/10.1007/s00521-020-04708-x.

[23] S.SMADI,M.ALAUTHMAN,O.ALMOMANI, A. SAAIDAH, F. ALZOBI, Application layer denial of services attack detection based on stacknet, International Journal of Advanced Trends in Computer Science and Engineering 3929 (3936) (2020) 2278–3091.doi:https://doi.org/10.30534/ijatcse/2020/215932020.

[24] S. Smadi, N. Aslam, L. Zhang, Detection of online phishing email using dynamic evolving neural network based on reinforcement learning, Decision Support Systems 107 (2018) 88–102. doi:https://doi.org/10.1016/j.dss.2018.01. 001.

[25] R. Zhang, B. Li, B. Jiao, Application of xgboost algorithm in bearing fault diagnosis, in: IOP Conference Series: Materials Science and Engineering, Vol. 490, IOP Publishing, 2019, p. 072062. doi:doi:10.1088/1757-899X/490/7/072062.

[26] N. N. Thilakarathne, An evaluation of machine learning classi-fiers for prediction of attacks to secure green iotinfrastructure,International Journal of Emerging Trends in Engineering Research 9 (5).