Journal of Theoretical and Applied Information Technology
15th August 2020. Vol.98. No 15

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

P2P BOT DETECTION USING DEEP LEARNING WITH TRAFFIC REDUCTION SCHEMA

MOHAMMAD ALAUTHMAN

Department of Internet Technology, Faculty of information technology, Zarqa University, Zarqa, Jordan.

E-mail: malauthman@zu.edu.jo

ABSTRACT

Nowadays, Botnet detection plays a vital role in assuring of information and internet security. This research introduces a scheme for peer-to-peer Botnet detection using a deep neural network in collaboration with the features selection approach. A classification and regression tree, ReliefF algorithm, and principal component analysis are utilized to choose the most significant features, and a deep neural network model is built based on adaptive learning methods (ADAM). The approach used in the proposed system utilizes network traffic alone, and the packet payload does not influence it, thus, avoids inherent shortcomings, such as the failure to handle encrypted payloads, as well as, preventing unknown malware from being addressed with rule-based antivirus software to be combated. This study compares the proposed model with classical machine learning methods like Gaussian NB, Logistic regression, SVM and Random Forest. The experiment results note that the proposed deep neural network model achieve the best performance with a principal component analysis as a features selection method. From the experimental results, the proposed method reached the accuracy of 98.2% along with 0.015 % false positive rate and 0.64 Root mean square error.

Keyword: *P2P Network, Botnet, Intrusion Detection, Deep Learning, Traffic Reduction.*

1. INTRODUCTION

Recently, the interesting of Botnet problems are increased in comparing with other computing threats due to the botnet is classified to be the riskiest way for preparing online crimes[1]. A Botnet network is consisting of Bots, Command and control channel (C2C) and Botmaster. A Bot infects the machines using several techniques such as backdoors, Trojan horses without the permission of the user. The Botmaster admin of the Botnet network via C2C in order to execute malicious activities [2-5]. However, for the commercial gain, Botmaster rented or sold Botnet networks in dark web. Then the attacker can start several kinds of malicious activity such as generating phishing web pages, stealing confidential users data, perform enormous volumes of spam emails and performing a DDoS attack[6].

According to a recent Spamhaus Botnet Threat Report 2017, The volume of such botnets "C2C" is grown by 32% in 2017 [7]. Symantec threat report 2016 indexed around 430 million different malware pieces[8]. Moreover, In May 2017, self-propagating malware such as WannaCry was infecting hundreds of thousands of computers over 150 countries within a few days[9].

Although the efforts stated in the literature to reduce the influences of Botnet activities, the variety of Botnet protocols and architecture make detection of Botnet is a critical job for the cybersecurity community[10-15].

Recently, deep Learning techniques are getting considerable attention due to the ability to deal with massive dataset size. Deep learning approach is investigated with many research problems such as image, sound, speech recognition, signal processing areas, natural language processing and Intrusion Detection System (IDS)

In this study, we develop a P2P Bot detection approach using deep neural networks incorporated with features selection techniques to identify computers connections that begin malicious traffic behaviors. The main contributions of this paper are given below:

- Adopting a deep neural network to detect the P2P Bot connections.
- Evaluating the performance of various features selection technique based on the deep neural network.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

- Comparing the performance of the proposed method with the classical machine-learning algorithm.

The remainder of the paper is structured as follows: In Section 2, a brief introduction about Botnet and classification of current detection approach. In Section. 3, the proposed technique is explained in details. Dataset and experimental results are discussed in Section 4. Finally, conclusion and future research direction are presented in section 5.

2. LITERATURE REVIEW

Botnets belong to a group of preprogrammed bots, controlled by a single Botmaster entity [16]. In order to perform multiple distributed and coordinated attacks remotely, the Botmasters have control of one or more C2C servers that are used to distribute commands to their bots. Bots must report back to the Botmaster the status of the action after these tasks have been executed [3, 4, 17]. This reporting is carried out via the C2C server.

The most crucial component in the Botnet life cycle is the mechanism of the C2C server. The Botmaster can communicate with Bots using the C2C mechanism. Furthermore, establishing the C2C channel represents the main difference between Botnets and traditional malware[18]. In contrast to popular types of malicious software, Botnet operates maliciously as a group of infected machines using a C2C-channel structure. The Botmaster can, therefore, use this channel to order the attack by thousands of Bots. Cooke and colleagues under their C2C mechanism ranked Botnets into three different groups: centralized, distributed and random. The first academic analysis of P2P Botnet [19]was also presented in the paper. Botnets were divided into four classes, including IRC, HTTP, P2P and Hybrid botnets [4] in terms of their development environments.

The P2P Bot after infecting the machine they will try to find other peers in order to join the Botnet network so they set-up a huge number of connections in order to find any peer still in life to receive updates and to give notes for the other peer in the same network. P2P Bots after infection stage need to communicate with the C2C server/peer to receive updates or more instructions from the Botmaster bypass peers, the new peer (infected host) should regularly reply to update the status and gather information about the network status.

2.1. Taxonomy of Botnet Detection

Several Botnet detection methods have been observed in recent years, which can be classified into four groups: Signature-based, Data-mining and DNS-based [20]. Other studies like Han et al. [21] have categorized the detection of P2P Botnet into three main groups: machine learning, data mining and analysis of network traffic. In addition, Zeidanloo and colleagues also listed the Botnet detection system as IDS or Honeynets and divided the IDS into three domains: specification-based, abnormality and signature-based. What is more, the installation point of the detection system like network-level, host-level and hybrid are also used to classify the botnet detection system[22]. Lu et al. classify of Botnet detection systems based on machine learning model as a supervised or unsupervised technique [23].

Supervised machine learning approaches are useful solutions for solving several classification jobs in various fields through utilizing a training dataset in order to build a classification.

Stevanovic and Pedersen [24]presented a Botnet detection based on the network flow features. A combination of 39 flow features extracted from flow packets is utilized to classify malicious network activity. Also, eight supervised machine learning algorithms including SVM, naive Bayesian classifier, decision tree, logistic regression and Bayesian network classifier are applied to detect Botnet activities. However, the best accuracy 95.7% was achieved by using a random forest algorithm based on ISOT dataset [25].

Kirubavathi and Anitha [26] proposed a behavioral Botnet detection approach using features of networks flow and machine learning techniques. The researcher extracted four features to describe the network traffic which includes: incoming and outgoing packets rates, first packet size and Bot response packet rates overall flow packets. The naive Bayesian classifier is applied to classify the network traffic and achieved 99% accuracy with 96.9% F-measure. However, the

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

four features are chosen based on the expert analysis of the botnet behavior. Moreover, the number of selected features may not enough to clearly show a Botnet connection, so, the reliability of the achieved results required to assessed using further network traffic properties in order to represent the majority of botnet behavior.

A hybrid approach has been adopted by the authors in [27] to classify and discriminate P2P-botnet traffic from P2P. In order to create three datasets, Storm, Zeus, Waledac and Conficker botnets traffic has been mixed. The approach consists of two steps: 1) signature-based approach to heuristic detection and 2): statistical classifier using the heuristic detection pattern. This classification is built based on the algorithm of the decision tree. The proposed approach provides low overhead and achieves 97.10 and 97.06 percent flow and byte accuracy with real data sets respectively.

Deep learning techniques such as restricted Boltzmann machines, Deep belief networks and auto-encoders are widely utilized for the intrusion detection system. For example, Tao, X. et al. [7] utilized features reduction approaches such as PCA, LDA, and Fisher score in order to remove redundant data from KDD Cup '99 dataset. Moreover, they combined Fisher score with the Deep autoencoder algorithm. However, The Deep autoencoder algorithm notably enhanced the accuracy of the classification rate for the network traffic. Moreover, they compare the performance of the proposed approach with J48, BPNN, and SVM algorithms in terms of classification accuracy.

In addition, recurrent neural networks are adopted for classification, various researchers have used the Gated Recurrent Unit (GRU) and Long-Short-Term-Memory algorithm (LSTM) for IDS.

Kim. J. et al.[28] adopted the LSTM algorithm with Gradient Descent Optimization in order to set up efficient IDS. The approach achieves 97.54% accuracy with 98.95% recall. The same team utilized the Gated Recurrent Unit on intrusion detection dataset. The approach achieves 98.65% accuracy with 97.06% recall[29].

In 2018, Pektaş & Acarma [30] introduced a deep learning Botnet detection approach using convolutional and recurrent neural. The ISOT [25] and CTU-13 [31] dataset are used to test the validity of the proposed approach. Moreover, the proposed deep learning approach achieve around 99.3% accuracy on botnet detection.

Despite the enormous number of introduced Botnet approach, the detection of Botnets is a demanding job for the Internet community, because the size of internet traffic is growing dramatically and the Botnets maker try to perform activities in stealthiness way. Therefore, there is a necessity for introducing a new solution to have adaptability and efficiency. In this research, we introduce solution using deep learning incorporated with features selection approach in order to achieve the adaptability and efficiency on Botnet detection.

3. PROPOSED APPROACH

The main aims of our method include, 1) detect insensible malicious activities; 2) detect bots without required to perform deep packet inspection; 2) detect bots in a short period; and 3) the method detect Bots without required to analyze the whole network traffic.

The main components of the proposed detection scheme are network traffic filtering, features extraction, features selection and deep neural network as shown in Figure 1.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

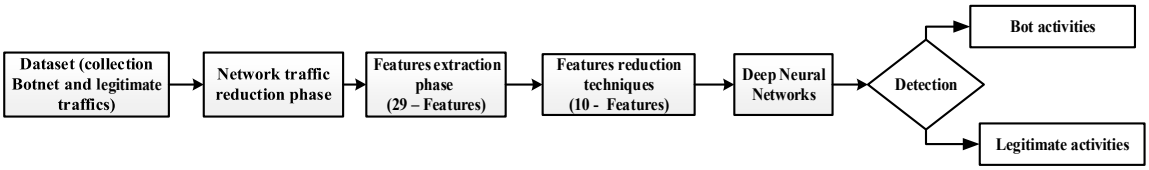

Figure 1. Phases of P2P Bot detection.

In network traffic reduction phase, we utilize our previous research work [16] to reduce the enter network traffic. The technique decreases the examiner network traffic through generating representative traffic of the entire network and avoiding Deep Packet Inspection such as most existing Botnet detection systems. In the feature extraction phase, we utilize our former extracted features set, which are assembled in 29-features based on 30-second of connections[16]. Classification and regression tree (CART) [32], Principal Component Analysis (PCA) [33] and ReliefF [34-36] are used as the feature selection method to reduce the numbers of ineffective stories. Table 1. shows the selected features using CART and ReliefF Algorithms.

Table 1: list of Features set by CART, ReliefF and PCA [16]

CART algorithm Feature	ReliefF algorithm Features	PCA
# of received control-pkt / flow.	# of sent RST–ACK packets / flow.	Linear combination of features
Avg. length of received control-pkt / flow.	# of sent FIN–ACK packets / flow.	
The ratio of avg. length of sent packets over the avg. length of control-pkt/ flow.	# of failed connection / flow.	
The total number of bytes / flow.	# of sent SYN packets / flow.	
Avg. length of control-pkt / flow.	The ratio of incoming control-pkt / flow.	
Avg. time between an attempt to create connection / flow	(transmitted SYN - received SYN-ACK)/ flow.	
Avg. length of transmitted control-pkt / flow.	Avg. time between an attempt to create connection / flow.	
# of control / flow.	The ratio of avg. length of sent packets over the avg. length of control-pkt/ flow.	
# of sent bytes / flow.	# of received FIN–ACK packets / flow.	
# of sent failed connection / flow.	# of transmitted SYN-ACK packets / flow.	

Finally, a deep neural network [37] with ADAM learning algorithm [38] is chosen as a malicious behavior detector because of its high accuracy and adaptive rates.

The detection approach based on the deep learning architecture concentrates on the design and training of the Deep Neural network (DNN) model. In this research, the proposed model principally comprised of the fully-connected layers with dropout layer. Figure 2. represents the structure of the DNN applied for Bot detection, which uses all 10 selected features. The input layer consists of 10 neurons (features) which are supplied to the DNN. The main component of the hidden layer is four fully connected layers'

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

contents (50 – 40 – 30 -20) neurons and dropout layer. Dropout is an effectual method proposed in [39] for enhancing error generalization and reducing overfitting cases for deep neural networks.

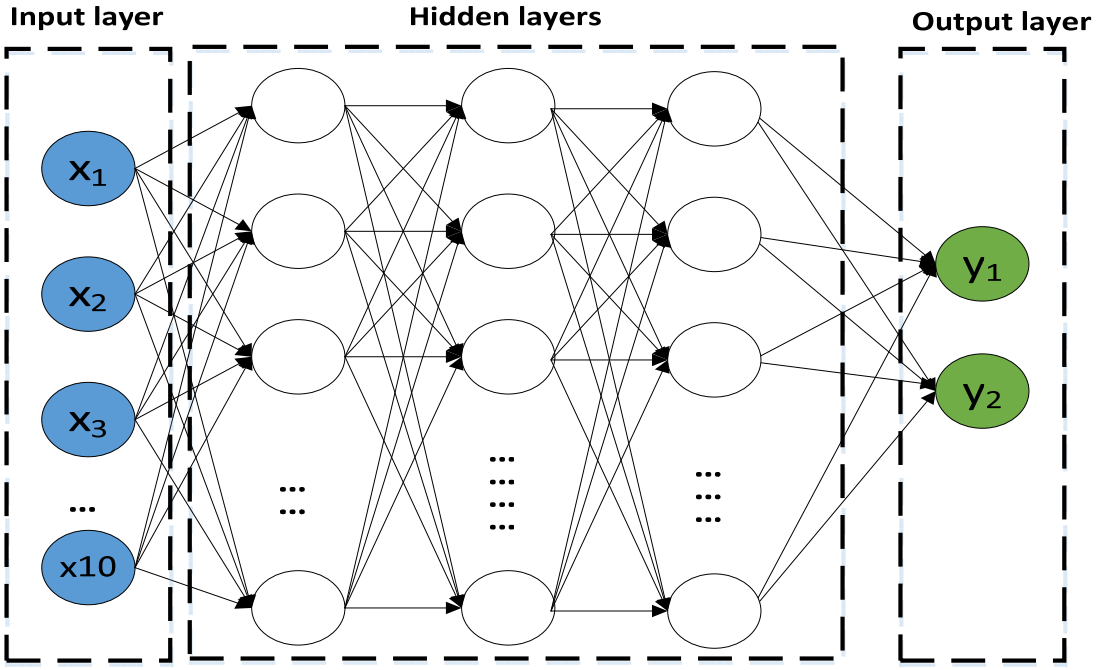

Figure 2. Structure of Deep Neural Network.

In addition, Rectified Linear Activation function(ReLU) [40] is utilized as activation function of the hidden layers. Because the *ReLu* is a non-linear function so that it can improve the model performance and it has the ability to handle the complicated classification cases. However, the function returns 0 if it receives any negative input, but for any positive value x, it returns that value back. So it can be written as $f(x) = max(0, x)$.

Adaptive moment estimation(Adam) [38] is used as an optimization technique. Adam achieves the advantages of both Adaptive Gradient Algorithm (AdaGrad) and Root Mean Square Propagation (RMSProp). In ADAM algorithm, the decaying average of past and past squared gradients are estimated as follows:

$$\boldsymbol{m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t} \qquad (1)$$

$$v_t = \beta_1 v_{t-1} + (1 - \beta_2) g_t^2 \qquad (2)$$

m_t (first moment) represents the decaying average of past gradients exponentially, v_t (second moment) represents the decaying average of past squared gradients exponentially and g_t denote the gradient at time step *t*. $m_t$and v_t are initialized as vectors of zeros ,and β_1 and β_2 are close to one. In addition, the biases are corrected by computing bias-corrected for the $m_t$and $v_t$as follows:

$$\tilde{\boldsymbol{m}}_{\boldsymbol{t}} = \frac{\boldsymbol{m_t}}{\boldsymbol{1 - \beta_1^t}} \qquad (3)$$

$$\tilde{v}_t = \frac{v_t}{1 - \beta_2^t} \qquad (4)$$

$\tilde{m}_t$ denotes the unbiased estimation of the first moment and $\tilde{v}_t$ denotes the unbiased estimation of the second moment.

Adam updates the rules as follows:

$$\boldsymbol{w\,(t+1) = w(t) - \frac{\alpha}{\sqrt{\tilde{v}_t} + \epsilon} \tilde{m}_t} \qquad (5)$$

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

α denotes the step rate parameter and ϵ represents the safety offset of division for the second moment, however, in the research we used 10^{-8} as the default value of ϵ.

4. EXPERIMENTS AND RESULTS

In this paper, we use the Tensorflow [28] as the backend of the KERAS [41], we construct a neural network model, which mainly applies deep neural networks using deep learning KERAS framework with ADAM optimizing function [38]. And The cost function is estimated based on a binary cross-entropy [42, 43] .

4.1. Dataset

We used the ISOT dataset [25] and ISCX dataset [44]. The ISOT dataset contains two types of network traffic malicious traffic includes: Waledac and Storm Bots and legitimate network traffic. The second is the ISCX dataset which contains legitimate and malicious network traffic. Table 2. Present the network traces utilized in the proposed model.

Table 2. Datasets.

Traffic type	Duration	Number of packets	Number of control packets
Storm Bot traffic	3115 seconds	128241	64551
Waledac Bot traffic	605 seconds	118379	49536
Normal traffic (ISCX)	9511 seconds	419659	165218
Normal traffic (LNBL)	126273 seconds	564999	166308

4.2. Results and performance evaluation

In order to evaluate the accuracy of deep learning model, the cross-validation technique is used to evaluate the quality of the proposed approach. N = 10 folds were chosen in the evaluation of our experiments. Several evaluation techniques such as false positive rate (FPR), detection rate (DR), accuracy (ACC), log loss, Area under ROC (AUC) and the F-Score are applied to our experimental results.

A classifier can identify a network connection into one of four categories[45-47]: 1) True Positive (TP), indicate the number of connection that correctly classified as Bot activities; 2) False Positive (FP), indicate the number of connections that incorrectly classified as Bot activities; 3) True Negative (TN), indicate the number of connections that correctly classified as legitimate activities; 4) False Negative (FN), indicate the number of connections that incorrectly classified as legitimate activities.

1. The *FPR* indicates the rate of legitimate connections incorrectly classified as Bot connections:

$$\boldsymbol{FPR = \frac{FP}{(TN + FP)}} \quad (6)$$

2. DR, also named recall, shows the rate of Bot connections that successfully identified as Bo.

$$\boldsymbol{DR = \frac{TP}{(TP + FN)}} \quad (7)$$

3. ACC shows the accurate predictions rate for all cases (legitimate and malicious).

$$\boldsymbol{ACC = \frac{(TP + TN)}{(TP + TN + FP + FN)}} \quad (8)$$

4. The F- score is a measure of a test's accuracy. However, F1-score reaches the best evaluation result at 1 and 0 represents the worst score.

$$\boldsymbol{F - score = \frac{(2 \times Precision \times Recall)}{(Precision + Recall)}} \quad (9)$$

5. RMSE shows the differentiation between the target label and the actual values computed by the detection system.

$$\boldsymbol{RMSE = \sqrt{\sum_{i=1}^{N} \frac{(yi - ti)^2}{N}}} \quad (10)$$

where $N$the sample is size, yi indecate outputs of the model and ti indicate targets of samples. Root Mean Square Error ($RMSE$) is one of the primary measures that show that variation between yi (model outputs) and ti (model targets), so, when $RMSE = 0$ denotes that the model prediction precisely matches the targets[48].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

6. The receiver operating characteristic (ROC) is a graphical chart that describes a classifier's performance. ROC curves plot the TPR on the vertical axis against the FPR on the horizontal axis. A sound classifier has an area under the ROC curve (AUC) nearby to 1.0. The AUC denotes the classifier's performance [49]. Moreover, the AUC is known to be a much more robust estimator of classifier performance [50].
7. Loss log: estimates the classification model performance using the probability of prediction output and the target labels [42].

$$L(\theta) = -\frac{1}{n}\sum_{i=1}^{n}|y_i\, log(\bar{y}_i) + (1-y_i)\, log(1-\bar{y}_i)| \quad (11)$$

$\bar{y}$ denotes the output of the neural network, y represent the real target of the neural network.

The results show that the technique archives the highest accuracy and detection rate with deep neural network and PCA features set at around 98 % and 97% respectively. The features set using the ReliefF algorithm gained lower detection and accuracy rates than the other methods at about 91% and 73 %, respectively, as presented in Table 3. Moreover, it can be clearly seen that the proposed scheme gives the highest F- score rates, MCC of about 96.9%, 95.6 respectively based on a PCA Feature set; meanwhile, the worst performance achieved with ReliefF features. Besides, the lowest FPR were achieved with ReliefF Features set.

Table 3. Deep Neural Network results with PCA, CART and ReliefF Features set

Evaluation	PCA	CART	ReliefF
ACC	0.9815 (+/- 0.0036)	0.9402 (+/- 0.0116)	0.9065 (+/- 0.0297)
DR	0.9736 (+/- 0.0123)	0.8946 (+/- 0.0610)	0.7287 (+/- 0.0701)
FPR	0.0152 (+/- 0.0040)	0.0401 (+/- 0.0133)	0.0168 (+/- 0.0321)
F1-Score	0.9693 (+/- 0.0061)	0.8991 (+/- 0.0257)	0.8237 (+/- 0.0593)
MCC	0.9561 (+/- 0.0087)	0.8587 (+/- 0.0277)	0.7773 (+/- 0.0723)

The standard deviation between folds' results was used to measure the results stability. As shown in Table 2 the PCA feature set obtained the best standard deviation for ACC, DR, FPR, F-score and MCC at 0.0036, 0.0123, 0.0040, 0.0061 and 0.0087 respectively. Meanwhile, the ReliefF feature set gave the worst standard deviation for ACC, DR, FPR, F-score, and MCC at 0.0297, 0.0701, 0.0321, 0.0593 and 0.0723 respectively.

The proposed scheme reached an average area under the ROC curve (AUC) of 0.995 for detection of bots based on PCA features set incorporated with the deep neural network. It is observed that our proposed model works sound in identifying of Bot malicious traffic. Besides, the lowest AUC were given with ReliefF Features set, as shown in Figure 3.

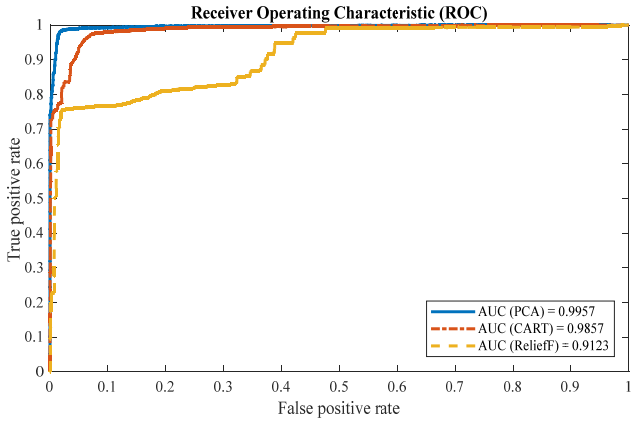

Figure 3. Area under ROC Comparison

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Afterward, the performances of the proposed scheme based on features selection algorithms were analyzed based on the average Log loss and RMSE, and the PCA technique reached the best Log loss and RMSE rates at around 0.6407 (+/- 0.1259) and 0.0167 (+/- 0.0016) respectively as shown in Figure 4(a). In addition, the worst Log loss and RMSE were given with ReliefF Features set as shown in Figure 4(b).

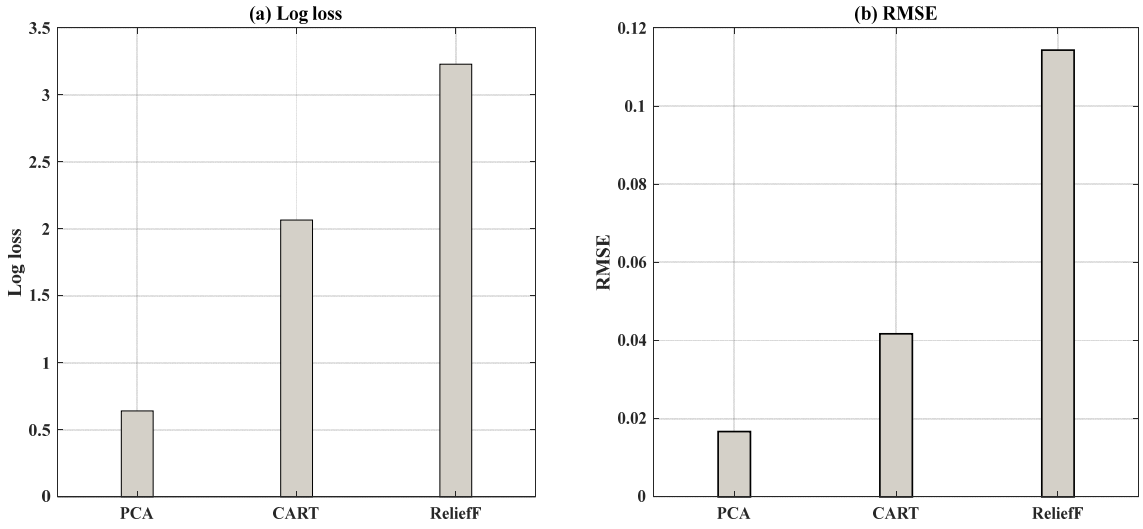

Figure 4. Log loss and RMSE Comparison

Table 4. Shows the comparison of the proposed approach performance with respect to some current research in the area of botnet detection. For example, Pektaş & Acarma [30], they used two types of neural networks: 1) Combination of recurrent neural and convolutional networks and 2) Stand-alone Dens network, furthermore, they achieved a 99.0% accuracy with 97.3% F-score in the first neural network type and 96.7% accuracy with 94.3% f-score in the second type. Although the authors achieved high accuracy and F-score in the combination of recurrent neural and convolutional networks, this solution it is time costly than our approach (sequential deep dens neural network with features section and Adam learning approach). In [26] the authors achieved 99% accuracy with 96.9% F-score using Naïve Naive Bayes algorithm based on ISOT dataset. However, they obtained good results but the number of selected featured (chosen based on the expert analysis) may not adequately to classify botnet behavior. In [24], the best accuracy 95.7% was achieved by using a random forest algorithm based on ISOT dataset. Our proposed approach outperforms by reaching 98.2% accuracy, 96.1% F1-score and 97.4% detection rate. The comparison results show that the proposed sequential deep neural network better botnet detection accuracy with a low false positive rate in two datasets in comparison with Stand-alone Dense network [30]. Furthermore, our approach achieved better F-score and detection rate than [24, 26].

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

Table 4. Comparison of the proposed schema with other techniques using the same dataset

Research work	Dataset	Technique	Performance evaluation
Kirubavathi and Anitha [26]	ISOT	Naïve Naive Bayes algorithm	Accuracy: 99.1% F-score: 96.9%
Stevanovic and Pedersen [24]	ISOT	Random forest algorithm	Accuracy: 95.7%
Pektaş and Acarman [30]	ISOT	Combination of recurrent neural and convolutional networks	Accuracy :99.0% F-score: 97.3%
		Stand-alone Dense network	Accuracy: 96.7% F-score: 94.3%
Our study	ISCX and ISOT	DNN	Accuracy: 98.2 % F-score: 96.93% Detection: 97.36%

We conduct our experiments using classical machine learning algorithms that used for Botnet detection such as Random forest [51], support vector machine (SVM) [52], logistic regression [53] and Gaussian Naive Bayes[54]. As shown in Table 5. our proposed approach based on Deep Neural network with features selection performs better than classical machine learning algorithm.

Table 5. Comparison of our results with the classical machine-learning algorithm.

Algorithm	CART algorithm Feature		ReliefF algorithm Features		PCA	
	Accuracy	RMSE	Accuracy	RMSE	Accuracy	RMSE
Logistic regression	0.925 (+/-0.001)	0.062 (+/-0.002)	0.885 (+/-0.002)	0.114 (+/-0.003)	0.849 (+/-0.005)	0.150 (+/-0.003)
Gaussian NB	0.554 (+/-0.001)	0.445 (+/-0.001)	0.620 (+/-0.005)	0.379 (+/-0.003)	0.763 (+/-0.005)	0.236 (+/-0.003)
SVM	0.937 (+/- (0.004)	0.002 (+/-0.001)	0.903 (+/- 0.001)	0.006 (+/-0.001)	0.944 (+/-0.001)	0.055 (+/-0.001)
Random Forest	0.949 (+/-0.010)	0.003 (+/-0.004)	0.918 (+/-0.005)	0.002 (+/-0.001)	0.957 (+/-0.003)	0.023 (+/-0.005)

However, our technique can identify Bots malicious behavior with high detection rates without required to analyze all network traffic and bypass the encrypted network traffic in order to detect malicious behavior as soon as. One of the limitations of the approach is that it only checks TCP traffic and ignored UDP traffic; therefore, the proposed method is unable to detect Bots that use UDP network packets.

5. CONCLUSION

Achieving efficient classification results of classifiers by utilizing a minimum number of features has been one of the significant research goals in the intrusion detection system. This study introduced a comparative of three feature selection methods: PCA, CART and ReliefF, and their interactions with deep neural network classifier in the context of P2P botnet detection. The proposed system is independent of both IP addresses and payload information. Our experimental results show that the most useful

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

features are based on PCA. The overall accuracy of 98.2% was achieved through this approach. The proposed model is an efficient, lightweight with high detection accuracy and less false positive rate. We plan to extend our approach by applying deep reinforcement learning approach with unsupervised learning in order to improve accuracy and performance over time.

ACKNOWLEDGEMENTS

This research is funded by the Deanship of Research and Graduate Studies in Zarqa University/Jordan.

REFERENCES

[1] M. Alauthman, N. Aslam, M. Al-kasassbeh, S. Khan, A. Al-Qerem, and K.-K. Raymond Choo, "An efficient reinforcement learning-based Botnet detection approach," *Journal of Network and Computer Applications,* vol. 150, p. 102479, 2020/01/15/ 2020, doi: https://doi.org/10.1016/j.jnca.2019.102479.

[2] K. Alieyan, A. Almomani, M. Anbar, M. Alauthman, R. Abdullah, and B. Gupta, "DNS rule-based schema to botnet detection," *Enterprise Information Systems,* pp. 1-20, 2019.

[3] M. Alauthman, "An efficient approach to online bot detection based on a reinforcement learning technique," PhD, Northumbria University, UK, 2016. [Online]. Available: http://nrl.northumbria.ac.uk/id/eprint/29617

[4] A. Almomani, M. Alauthman, N. Aslam, O. Dorgham, and M. Al-Refai, "BOTNET BEHAVIOUR AND DETECTION TECHNIQUES: A REVIEW," in *Computer and Cyber Security: Principles, Algorithm, Applications and Perspectives*. USA: CRC press, 2018.

[5] A. Almomani, M. Alauthman, O. Almomani, and F. Albalas, "A proposed framework for Botnet Spam-email Filtering using Neucube," in *18th The International Arab Conference on Information Technology (ACIT)*, Yassmine Hammamet, Tunisia, 22-24 December 2017 2017: IEEE.

[6] A. Rawashdeh, M. Alkasassbeh, and M. Al-Hawawreh, "An anomaly-based approach for DDoS attack detection in cloud environment," *International Journal of Computer Applications in Technology,* vol. 57, no. 4, pp. 312-324, 2018.

[7] X. Tao, D. Kong, Y. Wei, and Y. Wang, "A Big Network Traffic Data Fusion Approach Based on Fisher and Deep Auto-Encoder," *Information,* vol. 7, no. 2, p. 20, 2016. [Online]. Available: http://www.mdpi.com/2078-2489/7/2/20.

[8] P. Wood, B. Nahorney, K. Chandrasekar, S. Wallace, and K. Haley, "Symantec global internet security threat report," *White Paper, Symantec Enterprise Security,* vol. 21, 2016.

[9] W. A. Carter, "Forces Shaping the Cyber Threat Landscape for Financial Institutions," 2017.

[10] (2014). *Statement before the Senate Judiciary Committee, Subcommittee on Crime and Terrorism*. [Online] Available: https://www.fbi.gov/news/testimony/taking-down-botnets

[11] IBM, "IBM X-Force 2012 Trend and Risk Report," 2013.

[12] M. Alauthman, A. Almomani, M. Alweshah, W. Omoushd, and K. Alieyane, "Machine Learning for phishing Detection and Mitigation," *Machine Learning for Computer and Cyber Security: Principle, Algorithms, and Practices,* p. 26, 2019.

[13] A. Al-Qerem, B. M. Abutahoun, S. I. Nashwan, S. Shakhatreh, M. Alauthman, and A. Almomani, "Network-Based Detection of Mirai Botnet Using Machine Learning and Feature Selection Methods," in *Handbook of Research on Multimedia Cyber Security*: IGI Global, 2020, pp. 308-318.

[14] M. Al-kasassbeh and T. Khairallah, "Winning tactics with DNS tunnelling," *Network Security,* vol. 2019, no. 12, pp. 12-19, 2019.

[15] M. Al-Kasassbeh, M. Almseidin, K. Alrfou, and S. Kovacs, "Detection of IoT-botnet attacks using fuzzy rule interpolation," *Journal of Intelligent & Fuzzy Systems,* no. Preprint, pp. 1-11.

[16] M. Alauthaman, N. Aslam, L. Zhang, R. Alasem, and M. A. Hossain, "A P2P Botnet detection scheme based on decision tree and adaptive multilayer neural networks," *Neural Computing and Applications,* journal article pp. 1-14, 2016, doi: 10.1007/s00521-016-2564-5.

[17] A. Alnawasrah, A. Alnomani, F. Meziane, and M. Alauthman, "Fast flux botnet detection framework using Adaptive dynamic evolving spiking neural network algorithm," in *2018 9th International Conference on Information and*

Communication Systems (ICICS), 3-5 April 2018 2018.

[18] H. R. Zeidanloo, A. Bt Manaf, P. Vahdani, F. Tabatabaei, and M. Zamani, "Botnet detection based on traffic monitoring," presented at the International Conference on Networking and Information Technology (ICNIT), Manila,Philippines, 11-12 June 2010, 2010.

[19] E. Cooke, F. Jahanian, and D. McPherson, "The zombie roundup: Understanding, detecting, and disrupting botnets," in *Proceedings of the USENIX SRUTI Workshop*, 2005, vol. 39, p. 44.

[20] M. Feily, A. Shahrestani, and S. Ramadass, "A Survey of Botnet and Botnet Detection," in *Emerging Security Information, Systems and Technologies. SECURWARE '09. Third International Conference on*, 18-23 June 2009 2009: IEEE, pp. 268-273, doi: 10.1109/securware.2009.48.

[21] K.-S. Han and E. Im, "A Survey on P2P Botnet Detection," in *Proceedings of the International Conference on IT Convergence and Security 2011*, vol. 120, K. J. Kim and S. J. Ahn Eds., (Lecture Notes in Electrical Engineering: Springer Netherlands, 2012, ch. 56, pp. 589-593.

[22] H. R. Zeidanloo, M. J. Z. Shooshtari, P. V. Amoli, M. Safari, and M. Zamani, "A taxonomy of Botnet detection techniques," presented at the 3rd IEEE International Conference onComputer Science and Information Technology (ICCSIT), Chengdu, China, 9-11 July 2010, 2010.

[23] W. Lu, G. Rammidi, and A. A. Ghorbani, "Clustering botnet communication traffic based on n-gram feature selection," *Computer Communications,* vol. 34, no. 3, pp. 502-514, 2011, doi: doi:10.1016/j.comcom.2010.04.007.

[24] M. Stevanovic and J. M. Pedersen, "An efficient flow-based botnet detection using supervised machine learning," in *2014 International Conference on Computing, Networking and Communications (ICNC)*, 3-6 Feb. 2014 2014, pp. 797-801, doi: 10.1109/ICCNC.2014.6785439.

[25] S. Saad, Traore, I., Ghorbani, A., Sayed, B., Zhao, D., Lu, W., Felix, J., Hakimian, P., "Detecting P2P botnets through network behavior analysis and machine learning," presented at the Ninth Annual International Conference on Privacy, Security and Trust (PST), Montreal, QC, 19-21 July 2011, 2011.

[26] G. Kirubavathi and R. Anitha, "Botnet detection via mining of traffic flow characteristics," *Computers & Electrical Engineering,* vol. 50, pp. 91-101, 2016/02/01/ 2016, doi: https://doi.org/10.1016/j.compeleceng.2016.01.012.

[27] W. Ye and K. Cho, "P2P and P2P botnet traffic classification in two stages," *Soft Computing,* vol. 21, no. 5, pp. 1315-1326, 2017.

[28] T. T. H. Le, J. Kim, and H. Kim, "An Effective Intrusion Detection Classifier Using Long Short-Term Memory with Gradient Descent Optimization," in *2017 International Conference on Platform Technology and Service (PlatCon)*, 13-15 Feb. 2017 2017, pp. 1-6, doi: 10.1109/PlatCon.2017.7883684.

[29] J. Kim, J. Kim, and H. Kim, "An Approach to Build an Efficient Intrusion Detection Classifier," *Journal of Platform Technology,* vol. 3, no. 4, pp. 43-52, 2015.

[30] A. Pektaş and T. Acarman, "Deep learning to detect botnet via network flow summaries," *Neural Computing and Applications,* 2018/07/21 2018, doi: 10.1007/s00521-018-3595-x.

[31] S. García, M. Grill, J. Stiborek, and A. Zunino, "An empirical comparison of botnet detection methods," *Computers & Security,* vol. 45, no. 0, pp. 100-123, 9// 2014, doi: http://dx.doi.org/10.1016/j.cose.2014.05.011.

[32] L. Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone, "Classification and regression trees. Wadsworth & Brooks," *Monterey, CA,* 1984.

[33] I. Jolliffe, *Principal component analysis.* Wiley Online Library, 2005.

[34] K. Kira and L. A. Rendell, "The feature selection problem: traditional methods and a new algorithm," presented at the Proceedings of the tenth national conference on Artificial intelligence, San Jose, California, 1992.

[35] M. ALKASASSBEH, "An Empirical Evaluation For The Intrusion Detection Features Based On Machine Learning And Feature Selection Methods," *Journal of Theoretical and Applied Information Technology,* vol. 95, no. 22, 2017.

[36] M. Al-Kasassbeh, S. Mohammed, M. Alauthman, and A. Almomani, "Feature Selection Using a Machine Learning to Classify a Malware," in *Handbook of Computer Networks and Cyber Security: Principles and Paradigms*, B. B. Gupta, G. M. Perez, D. P. Agrawal, and D. Gupta Eds. Cham: Springer International Publishing, 2020, pp. 889-904.

[37] G. Huang, Z. Liu, K. Q. Weinberger, and L. van der Maaten, "Densely connected convolutional networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, vol. 1, no. 2, p. 3.

[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[39] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *The Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929-1958, 2014.

[40] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104-3112.

[41] Chollet F et al, "Keras (2017)," ed: GitHub, 2017.

[42] D. E. Booth, "The Cross-Entropy Method," ed: Taylor & Francis, 2008.

[43] I. Obeidat, N. Hamadneh, M. Alkasassbeh, M. Almseidin, and M. AlZubi, "Intensive Pre-Processing of KDD Cup 99 for Network Intrusion Classification Using Machine Learning Techniques," ed: International Association of Online Engineering, 2019.

[44] A. Shiravi, H. Shiravi, M. Tavallaee, and A. A. Ghorbani, "Toward developing a systematic approach to generate benchmark datasets for intrusion detection," *Computers & Security*, vol. 31, no. 3, pp. 357-374, 2012, doi: doi:http://dx.doi.org/10.1016/j.cose.2011.12.012.

[45] M. Almseidin, M. Alzubi, S. Kovacs, and M. Alkasassbeh, "Evaluation of machine learning algorithms for intrusion detection system," in *2017 IEEE 15th International Symposium on Intelligent Systems and Informatics (SISY)*, 14-16 Sept. 2017 2017, pp. 000277-000282, doi: 10.1109/SISY.2017.8080566.

[46] M. Alkasassbeh, "A Novel Hybrid Method for Network Anomaly Detection Based on Traffic Prediction and Change Point Detection," *arXiv preprint arXiv:1801.05309*, 2018.

[47] M. Alkasassbeh and M. Almseidin, "Machine learning methods for network intrusion detection," *arXiv preprint arXiv:1809.02610*, 2018.

[48] A. Almomani, B. B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, "A Survey of Phishing Email Filtering Techniques," *IEEE Communications Surveys & Tutorials*, vol. 15, no. 4, pp. 2070-2090, 2013, doi: 10.1109/SURV.2013.030713.00020.

[49] J. A. Swets, *Signal detection theory and ROC analysis in psychology and diagnostics: Collected papers*. Psychology Press, 2014.

[50] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874, 6// 2006, doi: 10.1016/j.patrec.2005.10.010.

[51] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5-32, 2001.

[52] N. Cristianini and J. Shawe-Taylor, *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[53] M. Schmidt, N. Le Roux, and F. Bach, "Minimizing finite sums with the stochastic average gradient," *Mathematical Programming*, vol. 162, no. 1, pp. 83-112, 2017/03/01 2017, doi: 10.1007/s10107-016-1030-6.

[54] C. Robert, *Machine learning, a probabilistic perspective*. Taylor & Francis, 2014.