IAES International Journal of Artificial Intelligence (IJ-AI)
Vol. 9, No. 4, December 2020, pp. 609~615
ISSN: 2252-8938, DOI: 10.11591/ijai.v9.i4.pp609-615

A multilayer perceptron artificial neural network approach for improving the accuracy of intrusion detection systems

Abdulrahman Jassam Mohammed[1], Muhanad Hameed Arif[2], and Ali Adil Ali[3]
[1,2]Directorate of Education in Diyala, Ministry of Education, Diyala, Iraq
[3]Ministry of Higher Education and Scientific Research, Baghdad, Iraq

Article Info

Article history:

Received Apr 22, 2020
Revised Jul 27, 2020
Accepted Sep 20, 2020

Keywords:

Artificial neural network
DDoS attacks
Intrusion detection system
Multilayer perceptron

ABSTRACT

Massive information has been transmitted through complicated network connections around the world. Thus, providing a protected information system has fully consideration of many private and governmental institutes to prevent the attackers. The attackers block the users to access a particular network service by sending a large amount of fake traffics. Therefore, this article demonstrates two-classification models for accurate intrusion detection system (IDS). The first model develops the artificial neural network (ANN) of multilayer perceptron (MLP) with one hidden layer (MLP1) based on distributed denial of service (DDoS). The MLP1 has 38 input nodes, 11 hidden nodes, and 5 output nodes. The training of the MLP1 model is implemented with NSL-KDD dataset that has 38 features and five types of requests. The MLP1 achieves detection accuracy of 95.6%. The second model MLP2 has two hidden layers. The improved MLP2 model with the same setup achieves an accuracy of 2.2% higher than the MLP1 model. The study shows that the MLP2 model provides high classification accuracy of different request types.

Corresponding Author:

Abdulrahman Jassam Mohammed
Directorate of Education in Diyala
Ministry of Education
32001, Diyala, Iraq
Email: abid1974js@gmail.com

1. INTRODUCTION

The hackers have attacked the cyber system every single day. The information security companies and governments of various countries have spanned a significant consideration to prevent the distributed denial of service (DDoS) attacks [1-2]. The conventional cyber a threat on the web serves as the DDoS. However, sending huge packets to the web servers from attackers' tools is called the denial of service attack (DoS). All of the network, transport, and application layers have been used ICMP, TCP, and HTTP protocols, respectively, to prevent the DoS attacks [3-4]. Figure 1 illustrates a basic topology of the DDoS attack. Whereas, the attacker controls a large number of servers that sending the packets to the victim. The attacks attempt to block legitimate users to access a particular network service by sending a large amount of fake traffics to the victim network continuously [5]. The hackers have utilized the Botnets to realize the aim of DDoS attacks. Any network is created by a host computer are Botnets, while that is managed by some attackers are the network formed by called botmasters [6]. It is noted that the attackers send a large number of requests to a system during a short time, which makes that system hanging. Thus, the DDoS attack takes a shorter time than the DoS. Therefore, an accurate and fast intrusion detection system (IDS) is strongly enquired.

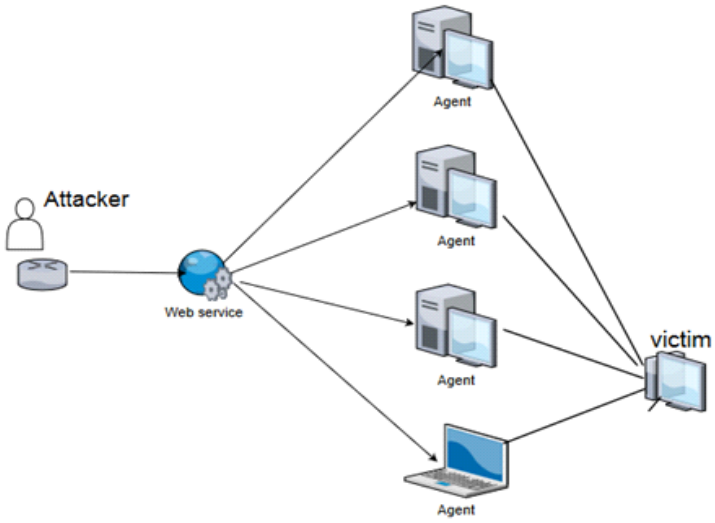


Figure 1. The architecture of DDoS attack

The mechanism of the IDS is integrated based on the classification of the host or network. The precision of classification the normal and anomaly requests gives the IDS accuracy. The classification system can recognize the DDoS attack based on traffics behavior. Many classification techniques were used to improve the IDS's accuracy. Machine-learning algorithms have been widely utilized for IDSs. Various artificial intelligence detection methods, namely, bayesian networks, [7] fuzzy logic, [8] genetic algorithms, [9] clustering, multilayer perceptron (MLP), [10] artificial neural networks (ANN), [11] software agent technology, and support vector machines (SVM), [12] are implemented for IDS application [2].

The ANN is one of the implemented techniques to prevent DDoS attacks. This section addresses a survey of the articles used ANN for IDS. The artificial neural network classification-based IDS was introduced. [13] The ANN model used NSL-KDD dataset with 29 features. The model achieved an accuracy of 81.2%. Then, Mane proposed ANN to detect the attack traffics which is misleading with the normal traffics [14]. Only 20000 samples of KDD99 dataset and (17 of 41) features were used to evaluate the performance of the proposed model. The system realized the accuracy of 98%. The selecting effective features for the ANN model that improved the system accuracy. Tsai in [15] proposed a time-delay neural network that can be used as an early detection system of DDoS attacks. This method is implemented based on the time parameter of each request. The system is tested a manually generated dataset. The system outcome displayed accuracy of 82%. In addition, it can detect a few types of attacks.

On the other hand, the MLP has been developed in several studies for IDS applications. The MLP includes multiple hidden layers to improve the learning rate of the neural network. Singh developed the MLP with a genetic algorithm to improve the detection efficiency of the IDS model [16]. The literature shows different types of the dataset that used to evaluate the IDS system, such as CIDA, DARBA FIFA world cup and KDD-NSL [2]. The selection of the effective features of the CAIDA 2007 dataset with genetic algorithm was improved the system performance in terms of DDoS classification. Wang used the MLP to overcome the DDOS attack problem [17]. Where the MLP was combined with dynamic features selection. During the training process, the model chose the optimal features according to the feedback checking of the errors of each epoch [18]. Therefore, an analyses investigation is recommended to highlight the effective detection method based on neural network algorithms.

This article provides two MLP models of ANN for classification-based IDS. The MLP based IDS system design with one hidden layer (MLP1) and two hidden layers (MLP2) to improve classification accuracy. The article is organized into several sections as follows. In section II, the design of IDS models is addressed and divided into three subsections, which are the methods and materials section (dataset, MLP1, and MLP2). After that, the simulation of the two proposed models and the evaluation of the results are addressed in Section 3. Finally, Section 4 concludes the study's achievements.

2. METHODS AND MATERIALS

In this section, brief backgrounds about the tested dataset the selected dataset's features are demonstrated. Moreover, the investigated ANN-based classification IDS methods are addressed.

2.1. Testing dataset

Among the different types of datasets, the NSL-KDD dataset is selected due to its variety of features that could be suitable to evaluate the performance of our proposed model. The available NSL-KDD dataset in [19-21] was utilized to evaluate the ANN model in this study. The NSL-KDD Dataset is the update version of 99KDD dataset [22-24]. The NSL-KDD was recorded according to a practical 125973 requests of a website in 2013. These requests were monitored and recorded with several features. The NSL-KDD dataset includes 42 features [2]. During the pre-processing, some of these features were excluded to meet the application requirements. Where the request type features were extradited, and then it is set as the target of the training IDS model. It is noted that the NSL-KDD dataset includes five types of requests, which are DOS of 45927, Probe of 52, R2L of 995, U2R of 11656, and Normal of 67343, as displayed in Figure 2.

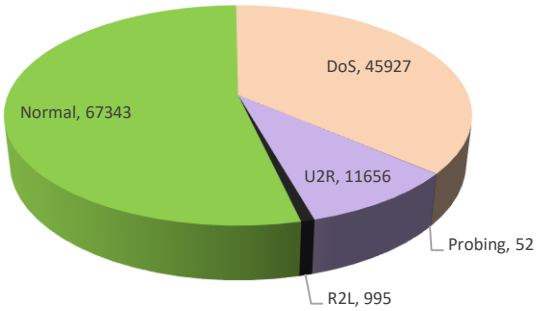


Figure 2. The request types of the NSL-KDD dataset

Some of the unwanted features, such as Flg, TCP and HTTP were removed from the dataset. Therefore, the remained 38 features of the selected dataset were imported as inputs of the IDS model. Specifically, 88181 samples of the dataset are used for training while 377912 samples are used for testing and validation.

2.2. The IDS framework

The ANN is modelled with one hidden layer (MLP1) and two hidden layers (MLP2) to improve classification accuracy. Where pre-pressing has selected the input and target features from the NSL-KDD dataset. In addition, 70% of the requests are imported to the ANN for the training process while the other 30% of the requests are used for testing. This training is implemented with several stopping criteria. For example, the KDD dataset is trained with 100 epochs. Also, the mean square error of 10-6 is set to stop the system training. After that, the bias and weights of the ANN are saved for testes process. The proposed ANN for IDS system is implemented by using MATLAB software. The IDS framework was designed to classify the request type according to the selected features, as shown in Figure 3.

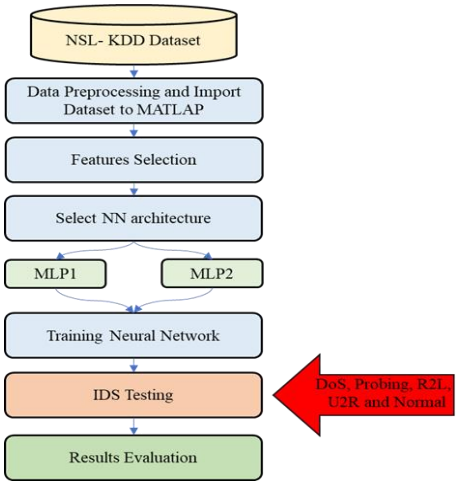


Figure 3. The IDS framework

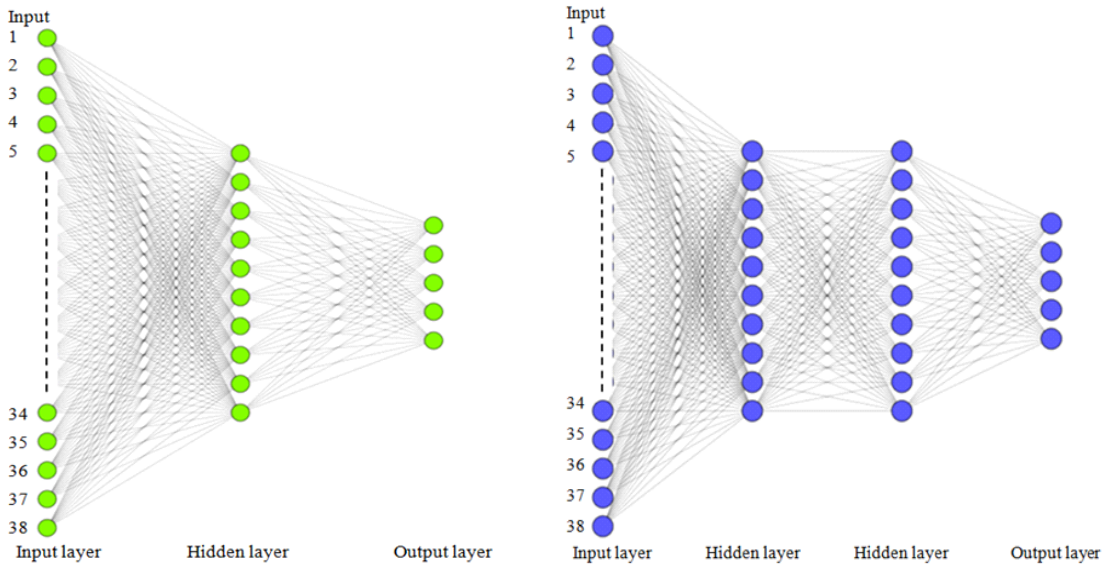
2.3. The MLP1 model

The ANN has been developed by using a computer model of multilayer perceptron with one hidden layer (MLP1). The MLP1 architecture is build based on the biological neuron. Where the information flows in the neurons changes the learning of the model based on its structure. The output of the MLP1 is used to check the error and update the model weight. The simple MLP1 configuration has three layers, which are input layer, hidden layer and output layer. The information flows through linkages among the layers. The output layer values are computed by multiplying the input information with the weight of the linkage according to the used activation function. The error can be calculated by comparing the output of MLP1 with the data target. This error is utilized to update the MLP1 weight by using backpropagation architecture . .Increasing the learning iterations meets the best weight and reduces the error

To design IDS, the MLP1 is implemented for the aureate classification system. The proposed MLP1 is developed and simulated by using MATLAB. The proposed MLP1 for IDS is designed with three layers, as displayed in Figure 4(a). The selected 38 features of NSL-KDD data are organized and used as inputs for the MLP1. Therefore, the input layer includes 38 input nodes one node for each input feature. The dataset includes so many requests. Each request has 38 features that imported to the design MLP1 as input. Whereas every single feature of a request is imported to an input node. The features of all the requests in the dataset are used to train the MLP1 respectively. Moreover, a single hidden layer is configured with 11 nodes. The hidden layer is activated by using the single-bias of each node. Hence, a sigmoid activation function is used to activate the neurons. The sigmoid is vertically normalized the training features between 0 and 1. The hidden layer is connected with the output layer of the ANN. Since the requests of the dataset are labelled into five different types, so the output layer is configured with 5 nodes. Therefore, the proposed MLP1 for IDS can be classified as requests into five types as shown in Figure 4(a). Each node of the hidden layer is connected with all the output nodes by using five neurons. More details about the implementation are addressed in the next paragraph.

2.4. The MLP2 model

Generally, the MLP2 has the same structure of the MLP1 except that multiple hidden layers are connected in series. The MLP2 includes a large number of inner connections of neurons to solve a problem. Several hidden layers are linked between the input layer and the output layer, as displayed in Figure 4(b). The input of the next layer is generated by applying an activation function to the output of the previous layer. The sigmoid activation function has widely used in the forward-propagation neural network training. The error is obtained by comparing the network output with the data target. The back-propagation adjusts the MLP2 weight vector. The discussed process for all the input vectors calls training epoch, which is repeated for several epochs until meeting the stop criteria.



(a) The MLP1 configuration of three-layers (b) The MLP2 configuration of four-layers

Figure 4. The MLP topology

The MLP2 is designed in this section to classify the NSL-KDD dataset for IDS application. The proposed PLM is designed with four layers (1 input, 2 hidden, and 1 output layers). The MLP2 layers' have the same number of nodes in the proposed ANN model. Where the input layer has 38 nodes, the hidden layers have 11 nodes and the output layer has 5 nodes. Figure 4 illustrates the MPL architecture with its four layers connection. Feed-forward back-propagation has connected the neurons among the layers' nodes. In addition, the sigmoid activation function is used with normalization the input features values between 0 and 1. Same the pre-processing that used for the ANN model is also implemented for this MLP2 model. Moreover, the sopping criteria (no of epochs= 100 and mean square error of 10-6) are used for the proposed MLP model. The request type of the tested dataset is set as a target while the selected 38 features are imported as inputs of the MLP IDS model. The model is implemented by using MATLAB for 100 epochs training. The weights and biases of the MLP2 are recorded for the evaluation process.

3. SIMULATION EVALUATION AND RESULTS

The MLP1 and MLP2 models are implemented by using MATLAB software, and 64 bit Windows 8. The used computer has specifications of Intel CPU core i7 @ 2.10 GHz with RAM of 4 GB. The performance of the system is computed by analyzing the results of the tested dataset.

The last 30% of the NSL-KDD dataset is used to test both the proposed model. Firstly, the proposed MLP1 with the recorded weights is implemented with the testing dataset. The output classification results of the tested MLP1 is recorded and compared with the target (request type feature). The confusion matrix is used to campout the classification accuracy of the MLP1 model, as illustrated in Figure 5. The MLP1 model achieves the best accuracy of 95.62% using (1) [25-31] after 100 epochs.

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

where TN, TP, FN, FP denotes true negative, true positive, false negative and false positive, respectively.

Secondly, the MLP2 model with the recorded weights is implemented by using the testing dataset. The outputs of MLP2 in IDS are recorded and validated with the target. The system accuracy is calculated by using (1). Figure 6 shows the confusion matrix of the proposed system. It can be noted that the MLP2 realized the best accuracy of 97.82% after 100 epochs.

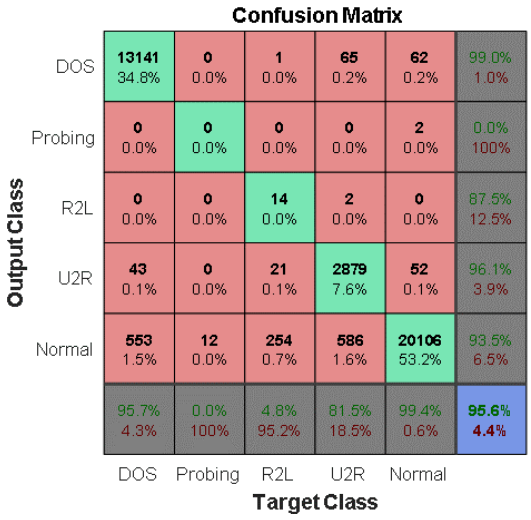


Figure 5. The confusion matrix and classification performance of the MLP1

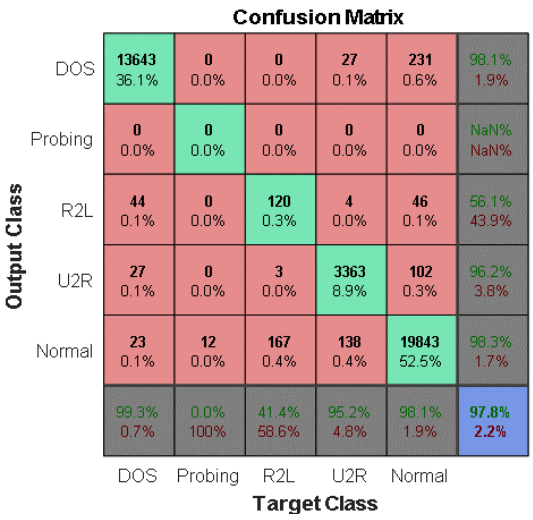


Figure 6. The confusion matrix and classification performance of the MLP2

Generally, the proposed MLP2 model demonstrates the accuracy of 97.82%, which is observed by 2.2% higher than the MLP1 model. Figure 7 shows the evaluation accuracy results for both MLP1 and MLP2 models.

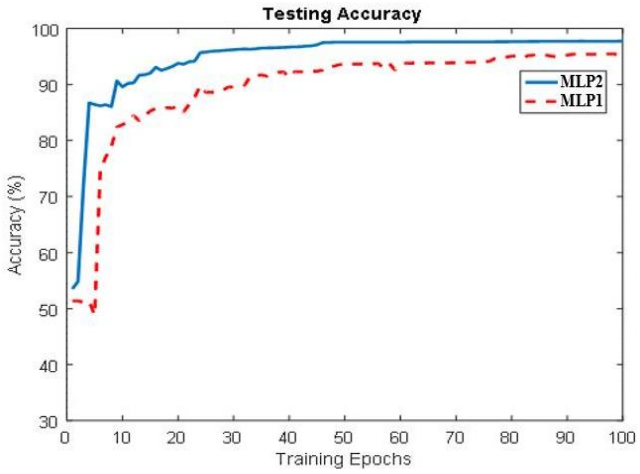


Figure 7. The accuracy evaluation between the MLP1 and MLP2 for 100 epochs

From the results, it can be argued that the additional hidden layer increased the learning rate more than the single hidden layer. In addition, both of the MLP1 and MLP2 are failed to detect the Probing request type. There are 12 Probing requests are implemented in the testing dataset. Therefore, a further investigation of the selected features and the developed IDS models is required.

4. CONCLUSION

In this article, two types of MLP are designed and modelled for IDS applications. The NSL-KDD dataset is used for both of the training and testing stages. Only 38 of 42 features are used for the classification stage. Same setup and parameters are used of both MLPs models. The evaluation study is demonstrated in this paper by the MLP2 model with two hidden layers achieves the accuracy of 97.82% while the MLP1 model achieves the accuracy of 95.62%. Subsequently, it is observed that increasing the number of hidden layers may improve the learning rate of ANNs. Utility, the proposed MLP-based IDS demonstrate a powerful tool for security applications. Further investigation is required in the features selection.

REFERENCES

[1] K. K. Choo, "The Cyber Threat Landscape: Challenges and Future Research Directions," *Computers & Security*, vol. 30, pp. 719-731, 2011.
[2] B.A. Khalaf, *et al.*, "Comprehensive Review of Artificial Intelligence and Statistical Approaches in Distributed Denial of Service Attack and Defense Methods," *IEEE Access*, vol. 7, pp. 51691-51713, 2019.
[3] G. A. Jaafar, *et al.*, "Review of Recent Detection Methods for Http DDoS Attack," *Journal of Computer Networks and Communications*, ID 1283472, pp. 1-10,2019.
[4] B.A. Khalaf, *et al.*, "An Adaptive Model for Detection and Prevention of Ddos and Flash Crowd Flooding Attacks," *IEEE International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 1-6, 2018.
[5] S.T. Zargar, *et al.*, "Survey of Defense Mechanisms against Distributed Denial of Service (DDoS) Flooding Attacks, " *IEEE communications surveys & tutorials*, vol. 15, pp. 2046-2069, 2013.
[6] S.S. Silva, *et al.*, "Botnets: A Survey," *Computer Networks*, vol. 57, pp. 378-403, 2013.
[7] A. Mehmood, *et al.*, "Nbc-Maids: Naïve Bayesian Classification Technique in Multi-Agent System-Enriched Ids for Securing Iot against Ddos Attacks," *The Journal of Supercomputing*, vol. 74, pp. 5156-5170, 2018.
[8] S. A. Mostafa, *et al.*, "A fuzzy logic control in adjustable autonomy of a multi-agent system for an automated elderly movement monitoring application," *International journal of medical informatics*, 112, 173-184, 2018.
[9] A. K. Malhi and S. Batra, "Genetic-Based Framework for Prevention of Masquerade and DDoS Attacks in Vehicular Ad-Hocnetworks," *Security and Communication Networks*, vol. 9, pp. 2612-2626, 2016.
[10] M. Wang, et al., "A Dynamic Mlp-Based Ddos Attack Detection Method Using Feature Selection and Feedback," *Computers & Security*, vol. 88, p. 101645, 2020.
[11] B. Gupta, and O.P. Badve, "Garch and Ann-Based Ddos Detection and Filtering in Cloud Computing Environment," *International Journal of Embedded Systems*, vol. 9, pp. 391-400, 2017.
[12] J. Ye, *et al.*, "A DDoS Attack Detection Method Based on SVM in Software Defined Network," *Security and Communication Networks*, ID 9804061pp. 1-8, 2018.
[13] B. Ingre and A. Yadav, "Performance Analysis of Nsl-Kdd Dataset Using Ann," *IEEE International Conference on Signal Processing and Communication Engineering Systems*, pp. 92-96, 2015.

[14] V. D. Mane and S. Pawar, "Anomaly Based Ids Using Backpropagation Neural Network," *International Journal of Computer Applications*, vol.136, pp. 29-34, 2016.
[15] C.-L. Tsai, *et al.,* "Early Warning System for Ddos Attacking Based on Multilayer Deployment of Time Delay Neural Network," *IEEE International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 704-707, 2010.
[16] K. J. Singh and T. De, "MLP - GA Based Algorithm to Detect Application Layer DDoS Attack," *Journal of information security and applications*, vol. 36, pp. 145-153, 2017.
[17] NSL-KDD dataset [online] available: https://www.unb.ca/cic/datasets/nsl.html, accessed Date Accessed on: 2009.
[18] Huang, *et al.,* "A distributed PSO–SVM hybrid system with feature selection and parameter optimization," *Applied soft computing*, vol. 8, pp. 1381-1391, 2008.
[19] M. Tavallaee, *et al.,* "A Detailed Analysis of the Kdd Cup 99 Data Set," *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pp.1-6, 2009.
[20] A. R. Wani, *et al.,* "Analysis and Detection of DDoS Attacks on Cloud Computing Environment Using Machine Learning Techniques," *IEEE Amity International Conference on Artificial Intelligence (AICAI)*, pp. 870-875, 2019.
[21] S. A. Mostafa, et al., "Evaluating the performance of three classification methods in diagnosis of Parkinson's disease, " *International Conference on Soft Computing and Data Mining*, pp. 43-52, Springer, Cham, 2018.
[22] J. A. Jupin, *et al.,* "Review of the machine learning methods in the classification of phishing attack,". *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 4, 1545-1555, 2019.
[23] S. A. Mostafa, *et al.,* "Social networking mobile apps framework for organizing and facilitating charitable and voluntary activities in Malaysia," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, 2020.
[24] M. A. Naagas, *et al.,* "Defense-through-Deception Network Security Model: Securing University Campus Network from DoS/DDoS Attack," *Bulletin of Electrical Engineering and Informatics*, vol. 7, no. 4, 593-600, 2018.
[25] B. A. Khalaf, *et al.,* "A Simulation Study of Syn Flood Attack In Cloud Computing Environment," *AUS journal*, vol. 26, pp. 1, 2020.
[26] M. A. Jubair, *et al.,* "A Survey of Multi-agent Systems and Case-Based Reasoning Integration," *International Symposium on Agent, Multi-Agent Systems and Robotics (ISAMSR)*, pp. 1-6. IEEE, 2018.
[27] M. A. Mohammed, *et al.,* "An anti-spam detection model for emails of multi-natural language," *Journal of Southwest Jiaotong University* 54, no. 3, 2019.
[28] M. A. Jubair, *et al.,* "Bat Optimized Link State Routing Protocol for Energy-Aware Mobile Ad-Hoc Networks," *Symmetry* 11, no. 11 (2019): 1409.
[29] M. A. Jubair, *et al.,* "Competitive Analysis of Single and Multi-Path Routing Protocols in Mobile Ad-Hoc Network," *Indonesian Journal of Electrical Engineering and Computer Science*, 14, no. 2, 2019.
[30] M. H. Hassan, *et al.,* "A Statistical Risk Assessment Method Of Dynamic Environments: A Case Study Of Air Pollution," *AUS journal*, vol. 26, pp. 1, 2020.
[31] A. S. Al-Khaleefa, *et al.,* "Feature Adaptive and Cyclic Dynamic Learning Based on Infinite Term Memory Extreme Learning Machine," *Applied Sciences*, vol 9, no. 5, pp: 895, 2019.