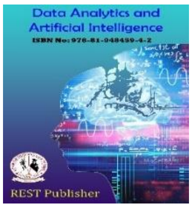
Data Analytics and Artificial Intelligence
Vol: 3(7), 2023
REST Publisher; ISBN: 978-81-948459-4-2
Website: http://restpublisher.com/book-series/daai/


Data Analytics and Artificial Intelligence
ISBN No: 978-81-948459-4-2
REST Publisher

Defend against DDoS attacks using optimized BPNN-APO in Cloud Computing

*** M. Arunadevi, V. Sathya**
Periyar University, Salem, Tamilnadu India
Adhiyamaan College of Engineering Hosur Tamilnadu, India
*Corresponding Author Email: asksmart84@gmail.com

Abstract: *One of the most harmful attacks on cloud computing is distributed denial of service (DDoS). By depleting resources, this attack renders cloud services unavailable to end customers, incurring significant financial and reputational damage. Therefore, creating defenses against this attack is essential for the broad adoption of cloud computing. This paper develops a new detection scheme-based back propagation neural network (BPNN) optimized by artificial plant optimization (APO). The proposed optimized detection method uses two benchmark datasets and four different performance measures for analyzing performance in experiments. Experimental results confirmed that the proposed detection scheme produced high detection accuracy and a faster convergence rate.*

Keywords: *DDoS attack detection, Artificial neural networks, Backpropagation neural networks, Artificial plant optimization, Cloud computing*

1. INTRODUCTION

A concept known as "cloud computing" allows users to access a variety of services on-demand with little interaction from either the cloud provider or the cloud user. It allows auto-scaling and is based on a utility-based pricing approach. Only the resources that the user uses must be paid for. When resources are auto scaled, they can be rapidly scaled up or down to meet the essentials of the user. The service level agreement between the cloud user provider determines when to scale up and when to scale down [1]. Security is just one of the issues this technology is facing. The security concerns with cloud computing are confidentiality, integrity, and availability. The availability of cloud services is impacted by a distributed denial of service (DDoS) attacks. The cloud's resources are down during this attack, making its services inaccessible. It is a form of denial of service attack in which numerous attackers target a single victim [2].These days, a variety of fields use machine learning (ML) techniques. The back propagation neural network (BPNN) is an ML algorithm that is applied to many real-world applications and also applied to detect DDoS attacks [3]. Because back propagation is seen as an efficient learning technique about the inputs and structure indicated before, it is the approach that is most frequently used. It also serves as a solid foundation for an accurate to detect DDoS attacks and is a widely used approach for artificial neural network (ANN) learning. However, Conventional BPNN algorithms have some drawbacks, such as the long training period, poor detection accuracy, and a high rate of false alarms. To overcome these problems, the present study focused on the artificial plant optimization (APO) algorithm used to optimize the parameters of BPNN. The objective of this study is to enhance BPNN’s capability so that it can produce a high detection rate with a minimal learning curve. The following are the research's contributions: The suggested APO-BPNN is used to identify DDoS attacks in a cloud environment. APO has adjusted the BPNN's settings to increase the detection rate and decrease the false alarm rate. Comparing the proposed approach to some benchmark detection algorithms.To model, create, and validate the detection system, two benchmark DDoS attack datasets are employed. The remaining sections are arranged as follows: Section 2 presents relevant papers; Sections 3 and 4 give descriptions of BPNN and APO, respectively; Section 5 discusses experimental comparisons, and Section 6 discusses the paper's findings.

2. BACK PROPAGATION NEURAL NETWORK

Rumelhart and McClelland's multi-layer feed-forward training model is called the BPNN [4]. The most widely used and straightforward neural network training approach is called BPNN, and it has a strong problem-ability [5]. The objective of the neural network architecture is to train the network to achieve stability between its capacity to respond to input patterns and its capacity to respond. Weights are calculated during the learning phase of the BPNN, a distinct neural network technique, to reduce error. In general, the multilayer perceptron struggles to efficiently determine the weight of the hidden layer. Because there are more hidden layers in the BPNN, the problem is more challenging. To update the architecture's weights, the error value is necessary. However, there are no explicit error values when changing the weights of the hidden layer. The total squared error of the output calculated by the networks is reduced using the gradient descent learning technique. The topology of a network is made up of several nodes connected by links, and it is organized into three layers: input layer, hidden layer(s), and output layer. The user's input signal is received by the nodes in the input layer, and the nodes in the output layer produce the desired model output. In this paper, a three-layer feed-forward neural network is used. Input values are $X = (x_1, x_2, ..., x_n)$ transferred to each of the hidden units $Z = (z_1, z_2, ..., z_k)$ after multiples with its weight then transferred to the hidden unit. Each hidden units estimate the result of the activation function and it sends its values to each output unit. The output unit estimates the results activation function to form the response of the net. Calculate the value of the hidden neuron by using

$$z_{-inj} = V_{oj} + \sum_{i=1}^{n} x_i v_{ij} \quad (1\ )$$

Calculate output neurons as follows,

$$y_{-ink} = W_{ok} + \sum_{j=1}^{p} z_i w_{jk} \quad (2)$$

Each output unit relates its calculated actual activation results with its target.

$$\delta_k = (t_k - y_k) f(y_{-ink}) \qquad (3)$$

Error information is considered by using the following method based on Equation 4.

$$\delta_j = \delta_{-inj} f(z_{-inj}) \qquad (4)$$

In conclusion, it can be seen that BPNN has good nonlinear performance, which makes it applicable for the simulation of nonlinear systems and suited for processing massive amounts of data in parallel. While training, the algorithm exhibits increased volatility and a slower rate of convergence, making it simple to settle into a local optimum.

3. ARTIFICIAL PLANT OPTIMIZATION

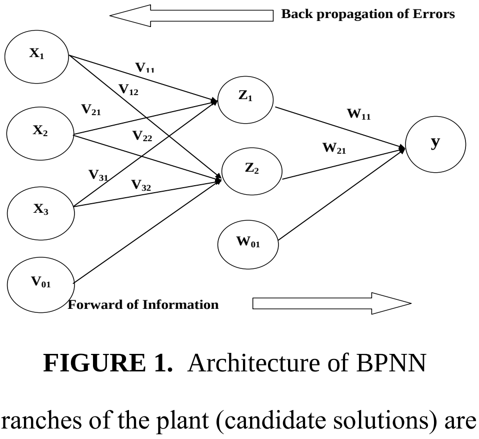


FIGURE 1. Architecture of BPNN

variable resources like sunlight. Branches of the plant (candidate solutions) are initialized, and their fitness value is computed utilizing operators like photosynthesis and phototropism. The amount of energy produced overall is quantified by photosynthesis, while phototropism shows how plants grow toward the sun. This shows how a potential solution progresses in the direction of an ideal one.

3.1 Photosynthesis :

Photosynthesis has to do with how a plant makes energy. In photosynthesis, the total amount of energy produced is determined by the photosynthetic rate. The following equation illustrates the quality of the produced energy when a rectangular hyperbolic model is applied.

$$r_i(t) = \frac{\mu If_i(t) R_{\max}}{\mu If_i(t) R_{\max}} - D_R \quad (5)$$

Where, $r_i(t)$ represents the photosynthetic rate of the i^{th} branch at time t. μ is the initial quantum's efficiency. The maximum net photosynthetic rate is denoted by $R_{\max}$. The dark respiratory rate is denoted by D_R . Both of $R_{\max}$ and D_R both regulate the rate of net photosynthetic. $If_i(t)$ is the light intensity which is represented as follows,

$$If_i(t) = \frac{f_{worst}(t) - f_i(t)}{f_{worst}(t) - f_{best}(t)}, \quad (6)$$

Where, $f_{best}(t)$ and $f_{worst}(t)$ are the best and worst intensity of light concerning time t respectively and $f_i(t)$ is the light intensity of branch i .

3.2 Phototropism: Phototropism is the term used to describe how plants grow in the same direction as the light source. Because they generate more energy, places with high light intensities are preferred in APOA. The branches are drawn to such places as a result. Branch i adopts the form depicted in Equation as follows,

$$g_i(t+1) = g_i(t) + C_p.F_i(t).rd(t) \quad (7)$$

Where, C_p is the parameter that denotes the rate of energy conversion. and $rd(t)$ is a random number which is a uniform distribution. $F_i(t)$ is the growing force for i^{th} branch which is calculated as follows,

$$F_i(t) = \frac{F_i^{total}}{\left\| g_i(t) - g_p(t) \right\|}.(g_i(t) - g_p(t)) \quad (8)$$

$\| \;\|$ is the Euclidean distance and F_i^{total} is determined as follows,

$$F_i^{total}(t) = \sum_{i \neq p} coeff.e^{-\dim P_i(t)} - e^{-\dim P_p(t)} \quad (9)$$

Where, dim is the problem dimensionality and $coeff$ is the controlling the direction of growth which is determined as follows,

$$coeff \begin{cases} 1 \;\; if \;\; P_i(t) > P_p(t) \\ -1 \; if \;\; P_i(t) < P_p(t) \\ 0 \qquad otherwise \end{cases} \quad (10)$$

The following equation includes the small probability P_m that reflects the influences of some random events:

$$x_i(t+1) = x_{\min} + \left(x_{\max} - x_{\min}\right).rd_1(), \;\; rd_1() \quad (11)$$

Where, $rd_1()$ and $rd_2()$ are two uniform distribution random numbers.

4. PROPOSED OPTIMIZED BPNN USING APO

The typical BP method can be improved by adjusting its parameters because it is simple to attain the local optimum due to improper parameter selection [7]. The major components of the overall optimization are the enhancement of connection weights and thresholds, the enhancement of the topology, and the enhancement of the learning parameters. Hence, the best solution of APO is considered as the initial parameters of the BPNN, which determine how the optimal individual is decoded into a set of connection weights and thresholds matching to the BPNN. Further, the training dataset can be used to determine the output error based on the initial parameters. The weight value and threshold will be changed and corrected if the termination condition is not satisfied, and the error will be handled using the BP. The network learning is stopped when the output error satisfies the necessary criteria, allowing for the creation of the final computation model.

5. EXPERIMENTAL ANALYSIS AND DISCUSSIONS

Experiments have been carried out to assess the performance of the suggested system. An Intel Core i5 CPU and 8 GB of RAM would be found on the system used for all testing. With the help of MATLAB 2015a, the experimental findings were produced. Comparing the effectiveness o f the proposed detection algorithms with various well-known detection algorithms, such as KNN, SVM, BPNN, GA-BPNN, PSO-BPNN, and APO-BPNN.

5.1 Dataset collection: The performance of our suggested system has been assessed using four different benchmark datasets, such as the NSL-KDD [8] the NSL-KDD dataset are categorized into four groups: DOS, R2L, U2R, and Probe. The samples in this collection include 41 features. Bruteforce, HTTP DoS, DDoS, and infiltrating attacks are among the four types of attacks in ISCX IDS 2012 dataset. There are 2,450,324 samples in this collection as a whole. Of these 2,450,324 samples, 2,381,414 are normal samples and 68,910 are assault samples. Only 2% of the samples in the dataset are attack samples; the remaining 98% are normal samples. This imbalance might harm the system's functionality. We performed tests using a subset of 68,910 attack samples and 70, 096 normal samples to avoid this

TABLE 1. Datasets Description

Datasets	Features	Training	Testing
NSL-KDD	41	20,000	5000
ISCX IDS 2012	19	19,200	4800

5.2 Pre-processing: The dataset contains a variety of feature ranges, most of which are simply too large. It is necessary in this case to use feature scaling, which is defined as the following, to reduce the range of a feature that ranges between 0 and 1.

$$x' = a + \frac{x - \min(x)(b-a)}{\max(x) - \min(x)} \quad (12)$$

5.3 Performance indicators:

In this study, a variety of performance metrics were used to assess how well the prediction algorithm performed. System performance was evaluated using Accuracy, Sensitivity, Specificity, and F-Measure (F1) measurements that were discovered in the literature. The following formulas were used to determine these metrics:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (13)$$

$$Sensitivity = \frac{TP}{TP + FN} \quad (14)$$

$$Specificity = \frac{TN}{FP + TN} \quad (15)$$

TABLE 2. Performance results for NSL-KDD dataset

Detection methods	Accuracy	Sensitivity	Specificity	Precision	F-Score
BPNN+APO	0.9827	0.9924	0.9927	0.9843	0.9825
BPNN+PSO	0.9795	0.9803	0.9795	0.9573	0.9618
BPNN+GA	0.9510	0.9682	0.9662	0.9428	0.9542
BPNN	0.8853	0.9264	0.9408	0.9081	0.9385
SVM	0.8385	0.8831	0.9361	0.8329	0.8937
KNN	0.7528	0.8094	0.8924	0.6528	0.8564

TABLE 3. Performance results for the ISCXIDS datasets

Detection methods	Accuracy	Sensitivity	Specificity	Precision	F-Score
BPNN+APO	0.9937	0.9864	0.9814	0.9972	0.9382
BPNN+PSO	0.9918	0.9817	0.9703	0.9864	0.9152
BPNN+GA	0.9735	0.9664	0.9476	0.9581	0.9038
BPNN	0.9464	0.9469	0.9168	0.9465	0.8747
SVM	0.8916	0.8828	0.8394	0.9364	0.8337
KNN	0.8357	0.7917	0.7549	0.8927	78.61

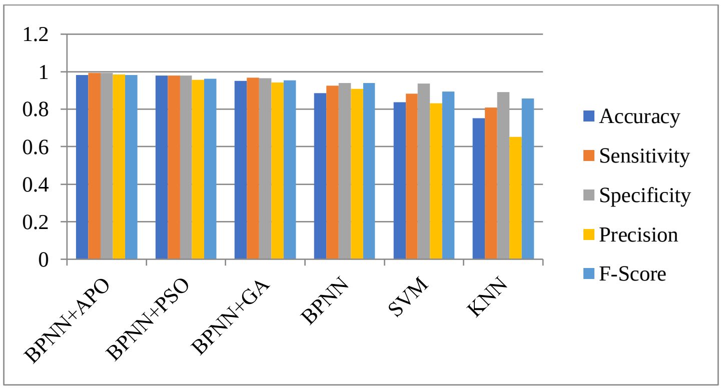


FIGURE 2. Performance result NSL-KDD dataset

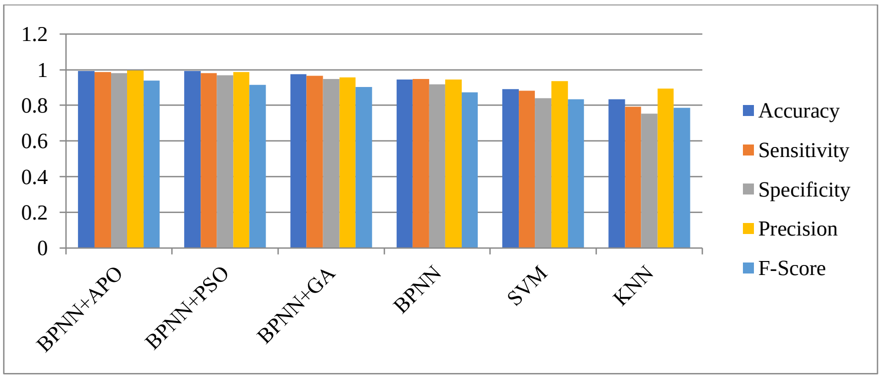


FIGURE 3. Performance results of the ISCX intrusion detection

According to this approach, true positive (TP) refers to attack traffic that was successfully predicted, true negative (TN) to normal traffic that was accurately predicted, false positive (FP) to attack traffic that was poorly predicted, and so on.

$$F - Score = 2 \times \frac{\Pr ecision \times Sensitivity}{\Pr ecision \times Sensitivity} \quad (16)$$

6. RESULTS AND DISCUSSIONS

Based on their performance metrics, the current section compares detection algorithms' performance outcomes for all datasets. The performance comparisons of several detection methods for NSL-KDD and ISCXIDS are shown in Tables 2 and 3, respectively. Figures 2 and 3 compare the effectiveness of several detection algorithms for the ISCXIDS and NSL-KDD, respectively. The experimental findings supported the idea that the optimized BPNN based on APO produces higher detection accuracy with a quick convergence rate. Additionally, the learning process for all datasets is taking a very short amount of time to finish.

7. CONCLUSION

In a cloud computing environment, an ML-based solution has been proposed to identify DDoS attacks. The detection system was created using BPNN optimized APO. The performance of the proposed detection scheme compared with some well-known detection schemes such as KNN, SVM, BPNN, GA-BPNN, PSO-BPNN, and APO-BPNN. The experimental results analysis shows that the performance of proposed detection scheme is better than other compared methods.

REFERENCES

[1]. G. Somani, M. S. Gaur, D. Sanghi, M. Conti, and R. Buyya, "DDoS attacks in cloud computing: Issues, taxonomy, and future directions," Computer Communications, vol. 107, pp. 30-48, 2017.

[2]. A. Rezaeipanah, S. E. Mousavipoor, M. Asayeshjoo, and M. Sadeghzadeh, "Combining Particle Swarm Optimization and Entropy to Detect DDoS Attacks in the Cloud Computing," Journal of Business Data Science Research, vol. 1, no. 1, pp. 33-43, 2021.

[3]. A. Aljuhani, "Machine learning approaches for combating distributed denial of service attacks in modern networking environments," IEEE Access, vol. 9, pp. 42236-42264, 2021.

[4]. D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," Cognitive modeling, vol. 5, no. 3, p. 1, 1988.

[5]. W. Dai, J.-Y. Wu, and C.-J. Lu, "Combining nonlinear independent component analysis and neural network for the prediction of Asian stock market indexes," Expert systems with applications, vol. 39, no. 4, pp. 4444-4452, 2012.

[6]. Z. Cui and X. Cai, "Artificial plant optimization algorithm," in Swarm Intelligence and Bio-Inspired Computation: Elsevier, 2013, pp. 351-365.

[7]. J. Zhang and S. Qu, "Optimization of backpropagation neural network under the adaptive genetic algorithm," Complexity, vol. 2021, 2021.

[8]. M. Tavallaee, E. Bagheri, W. Lu, and A. A. Ghorbani, "A detailed analysis of the KDD CUP 99 data set," in 2009 IEEE symposium on computational intelligence for security and defense applications, 2009: Ieee, pp. 1-6.

[9]. 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.