Detection of Intruder in Cloud Computing Environment using Swarm Inspired based Neural Network

Nishika[1]
Ph.D. Research Scholar
UIET, Maharshi Dayanand University
Rohtak, Haryana, India

Kamna Solanki[2]
Assistant Professor
UIET, Maharshi Dayanand University
Rohtak, Haryana, India

Sandeep Dalal[3]
Assistant Professor
DCSA, Maharshi Dayanand University, Rohtak, Haryana, India

***Abstract*—Cloud computing services offered a resource pool with a wide range of storage for large amounts of data. Cloud services are generally used as a demand-driven private or open data forum, and the increase in use has led to security concerns. Therefore, it is necessary to design an accurate Intrusion Detection System (IDS) to identify the suspected node in the cloud computing environment. This is possible by monitoring network traffic so that the quality of service and performance of the system can be maintained. Several researchers have worked on designing valid IDS with the help of a machine learning approach. A single classification algorithm seems to be impossible to detect intruders with high accuracy. Therefore, a hybrid approach is presented. This approach is a combination of Cuckoo Search. CS as an optimization algorithm and Feed Forward Back Propagation Neural Network (FFBPNN) as a multi-class classification approach. The user's request to access cloud data is collected and essential features are selected using CS as an optimization approach. The selected features are used to train FFBPNN with reduced training time and complexity. The experimental analysis has been performed in terms of precision, recall, F-measure, and accuracy. The evaluated value for parameters i.e., precision (85.5%), recall (86.4%), F-measure (85.9%), and accuracy (86.22%) are observed. At last, the parameters are also compared with the existing approach.**

Keywords*—Cloud computing; intrusion detection system; cuckoo search; feed forward back propagation neural network (FFBPNN)

I. Introduction

In this modern era, cloud computing has transformed the IT world with rapidly evolving and extensively accepted computing-based systems. The attractive features of Cloud Computing continue to increase integration in many sectors, such as governments, private, including industry, education, and entertainment [1]. According to the National Institute of Standards and Technology (NIST), cloud computing is defined as the computational model, which delivered services on-demand [2]. Cloud Computing provides a variety of applications and services to customers or users on the Internet. Services are provided remotely from various servers or the cloud, which is far from the users. Cloud Computing allows the user to use different software types in the cloud without installing the user system. Currently, there is a growing demand for clouds due to these devices, which leads to the need to take security measures because, with the increase in the demand, more security is required against threats [3]. There are already many businesses that use cloud computing services with their attractive features such as on-demand services, extensive network access, fast flexibility, and, finally, measurable services. Such features will allow users to focus on business processes while managing computing resources through a cloud service provider (CSP). Using cloud features, the operating costs are reduced by ensuring the compatibility and availability of different computing sources, simplifying device installation, and process with software and hardware updates [4]. There are several service provider models such as the private model, public model, community model, and hybrid model in the market. The cloud model that offers services to their individual or specific users is termed a public cloud. By using this cloud model, the services are delivered to the general public, managed, and controlled by private or government, or semi-government agencies. The private cloud infrastructure is extensive and provided for use by a single organization. The community cloud model is also presented to be used by a defined community. Here, community means a group of organizations with similar interests. A new cloud infrastructure is a hybrid cloud that consists of the right combination of different infrastructures, which can be individual (private), public, or community [5].

The organization, as well as the security of these cloud models, needs to be improved so that the stored data by many cloud users remain safe. This is possible through the utilization of the Intrusion Detection System (IDS). A suspicious entry in the network is known as an intrusion [6]. Therefore, it is necessary to design efficient IDS that can protect the stored data against suspicious users. IDS can be divided into two types, one is Host-based IDS (H-IDS), and the other is Network-based IDS (N-IDS) [7]. The first H-IDS intrusion detection program was developed using the original target system as the primary host computer, where some external interactions are often absent. HIDS will operate based on information collected using a personal computer system. It monitors all incoming and outgoing packets on the computer system and notifies users or the administrator if it is observed that there is a suspicious activity. This can be used commonly to protect personal information, which is valuable for several server-based systems [8].

NIDS will capture all network traffic and be further analyzed to detect all possible intrusions, such as port scans or sometimes Denial of Service (DoS) attacks. NIDS usually performs this detection by effectively processing IP and transport layer headers for all collected network packets simultaneously [9]. Network packets are collected in the presence of an anomaly, and a link will be obtained along with signatures for numerous notorious attacks, and this is used to compare user behaviors with their known profiles. Many hosts are working within a network that is protected from the attacker using the NIDS model. If it is to be run, one must come to know the location of the NIDS that is usually hidden [10]. An IDS is required to classify the malicious nodes in the deployed network. In the presented paper, the author has proposed an improved CS-FFBPNN based IDS system to protect the network from intruders. The authors considered to remove all inappropriate information from training data to increase classification accuracy and provide efficient IDS.

The major purpose of the proposed work is to provide a suitable intrusion detection system (IDS) to detect the suspected node based on the cloud environment. The researcher focused on the two types of attacks detection that are DDoS and Benign.

A. Contribution of the Work

The major contributions of the paper are discussed below:

- Identification of the suspected node in a cloud environment by designing an appropriate Intrusion detection system.
- Apply cuckoo search optimization algorithm for feature selection.
- Apply FFBPNN after feature selection to train the data.
- Design a hybrid approach by integrating cuckoo search algorithm and Feed Forward Back Propagation Neural Network.

II. Related Work

Cloud Computing techniques have become more sensitive as the services are provided in different parts of the world. Hashizume et al. (2013) have mentioned all possible attacks that appear in the cloud environment. They also tried to make recommendations to avoid risks and vulnerabilities [11]. Ghosh et al. (2015) have presented an efficient and fast IDS system for cloud networks combined with N-IDS and H-IDS. Using this approach, the collected network packets are analyzed, and then acknowledgment has been sent to the cloud administrator. K-nearest neighbor (K-NN) and neural networks have been used in combination to train and test the performance of the NSL-KDD dataset. The CSP has created a list of malicious IP addresses and then store that list for future use. The model has been designed to handle large data flow and generate reports accordingly [12]. Pandeeswari and Kumar (2016) have proposed an anomaly detection-based system by integrating the fuzzy c-means clustering algorithm to the ANN approach and named the designed method FCM-ANN. The performance of the intended approach has been compared with the Naïve Bayes approach and simple ANN approach using DARPA's KDD cup dataset 1999. Based on the experiments, it has been concluded that the designed FCM-ANN approach performed well with a high detection rate and a low false alarm rate [13]. Baig et al. (2017) have presented a multi-class ID system combined with the ensemble-based ANN approach to monitor the network traffic of the computer system. The designed system learned the behavior of suspected and regular users by cascading many NN (Neural Networks), and each network is trained using a small training dataset [14].

The small samples of training data have been prepared using a filter, and the system is trained using ANN with a boosting-based learning approach. The designed network has been performed better by comparing the results with the KDD CUP 199 dataset as well as with the UNSW-NB15 dataset. Mahajan et al. (2017) have presented a signature-based IDS approach to detect the anomaly behavior of data based on the traffic data flow. Results show that about 20 to 25 percent of the load has been increased on the CPU in contrast to usual traffic. Also, the CPU load during the virtual networking scenario of about 30% has been observed compared to the general networking case. It has also been concluded that the virtual network is failed to handle high-speed traffic [15].

Deshpande et al. (2018) have presented a host-based IDS model for the cloud environment. The designed model has alerted the cloud user about the suspected activities by observing the system call traces. The model has traced only the system call traces and the failed call traces instead of discovering all the system calls. The designed approach reduced the CPU burden to a great extent using security features and provided cloud security with an average accuracy of about 96% [16].

Hajimirzaei and Navimipour, (2019) have presented a novel IDS system that is cascaded with a multi-layer perceptron (MLP) network, Fuzzy clustering approach, and Artificial Bee Colony (ABC) as an optimization approach. The MLP was trained using the optimized values, which were later used to adjust the weight and bias of the MLP network. The performance has been analyzed based on different parameters to test the performance of the proposed system using the NSL-KDD dataset. Artificial Intelligence (AI) has already been included in this study to detect any interference in a particular cloud environment, and as a result of the application of AI techniques, this self-regulating IDA has been tested using real-time data with high network speed. This algorithm has been used to set up a highly reliable private cloud for the military and banking sector to monitor network activity [17].

Kumari et al. (2017) have presented Particle Swarm Optimization (PSO) in addition to the Genetic Algorithm (GA) as an evolutionary approach to secure data in the cloud network. Cloudsim toolkit has been used to evaluate the performance of the proposed work. The model presented improved results with reduced computation cost [18].

Manickam and Rajagopalan, (2019) have introduced a hybrid structure to prevent cloud networks using Glow Swarm Optimization (GSO) in integration to the Tabu Search (TS) approach. These techniques have been used to minimize

convergence time and resolve the problem of local optima [19].

Velliangiri *et al*. (2020) have focused on detecting DDoS attacks using a deep learning classifier. The collected log information from the number of users has been collected and retained in the log file. An appropriate feature from the log file has been gathered using Bhattacharya distance and then used to train the system using a deep network. Also, to obtained exact and proper features, the Taylor-Elephant Herd optimization-based Deep Belief Network (TEHO-DBN) has been modified using "Taylor Elephant Herd" as an optimization approach. The trained network is then used to detect DDoS attacks with an accuracy of 83%.

The adversary in the node accesses secret information, which may lead to eavesdropping or DDoS attacks. The presence of attack in the cloud environment may cause system overload and dumps packet that results in information losses. The primary attribute that is affected by the DDoS attack is IP address spoofing. The DDoS attack is an intruder that dumps the packets and exhausts the resources of an individual by sending huge data traffic towards the legitimate user. Therefore, it is essential to design a protection system that can detect the intruder and increase the system's performance [20].

III. Methodology Used

The Cuckoo Search (CS) technique as optimization and FFBPNN as a multi-class algorithm are used in this research. A detailed description of both methods is provided below.

A. Cuckoo Search

The brooding parasitism of cuckoo species inspires the CS algorithm. CS is a Swarm Intelligence (SI) approach invented in 2009 by Yang and Deb. CS mainly follows three rule sets:

- One egg is laid by each Cuckoo and dumped that egg in an arbitrarily selected nest.
- The high-quality egg is selected and is responsible for the next generation process.
- The host nest is static, and the egg laid by the Cuckoo is discovered by the host bird with a probability p∈ (0, 1). Depending upon the egg's quality, the host bird either select that egg or abandon the nest and create a new one [21].

As per the above rules, the CS works in the following way. Each egg in the given nest represents a candidate solution. Thus, each nest is supposed to be consists of one egg, although each nest may contain many eggs, generally represents a solution package. CS aims to create a solution in a better way. This is possible by replacing the worst cases with a current new solution presented in the nest. The selection of an appropriate egg is performed based on the objective function. In this research, the aim of using CS is to optimize the data set and hence reduce the searching time [22]. The workflow of CS is shown in Fig. 1.

Generally, the objective function is decided related to the problem with its minimum and maximum limit. Mathematically, the relationship between the minimum and maximum problem can be given by equation (1).

$$min(f(x)) = max(-f(x)) \quad (1)$$

After determining the minimum and maximum limit of an objective function, an additional parameter (p) called the transition probability is determined that measures the change in a new randomly produced nest. This parameter includes two elements of the CS algorithm: exploration and exploitation [23]. The extensive exploitation means premature convergence, whereas considerable investigation results in a decrease the convergence.

A random index egg is selected using the CS algorithm, and then these selected eggs are cross-checked against the fitness function of CS. If the attribute value does not satisfy the fitness function, then the attribute value is changed by equation (2).

$$Modified\ attribute\ value = \frac{\sum_{i=1}^{n} A_v}{n} \quad (2)$$

B. FFBPNN

FFBPNN is a parallel processing network, which is composed of a large number of simple integrated processors. This is one of the most commonly used artificial intelligence schemes. The three-layer structure of FFBPNN is shown in Fig. 2.

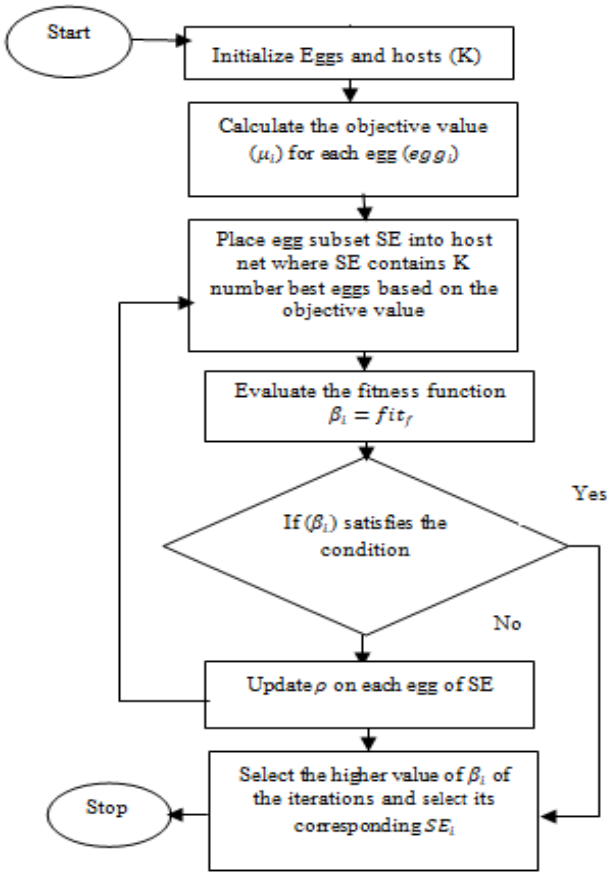

Fig. 1. The Flow of the CS Algorithm.

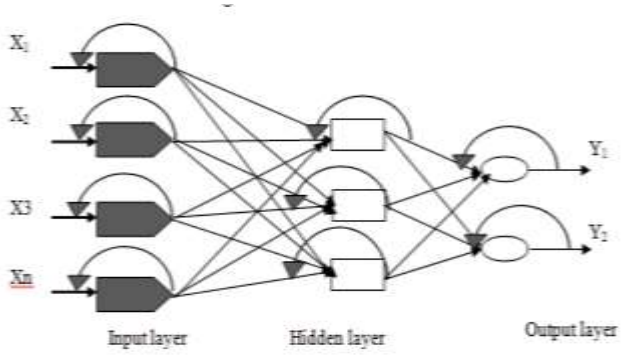

Fig. 2. Architecture of FFBPNN.

As shown in Fig. 2, input data {X=X1, X2, X3….. Xn} is passed as optimized data features to the input layer of FFBPNN. The arrow shows that the error value generated at the output layer is passed to the hidden layer to modify the weight matrix [24].

The standard multi-layer feed-forward network, including three layers, is depicted in Fig. 2. The architecture of the FFBPNN shares a similar characteristic having one layer, which is connected to its nearest layers and sharing neurons in a bidirectional manner. Here, bidirectional means that the information is transmitted or received in both directions. Every connection is assigned with some weight, which can be handled as per certain learning rules.

The generated error, which needs to be minimized, can be defined by equation (3).

$$Error\ (m) = \frac{1}{2}\sum_{r=1}^{r}[O_r - y_r]^2 \quad (3)$$

Where, O_r denotes the ground-truth value or label value. y_r represents the classified output during the simulation. Based on the generated error, the weight of the matrix in the hidden layer is changed as per equation (4).

$$\Delta w_{ij} = -K\frac{derror}{dw_{ij}} \quad (4)$$

IV. Proposed Work

In this research, an automatic intrusion detection system based on the machine learning approach in hybridization with an optimization approach named CS-FFBPNN has been presented. The integrated approach takes the advantage of the FFBPNN as machine learning and CS as a nature-inspired algorithm. CS is used due to its simplicity and ability to resolve non-linear real-world problems [22].

FFBPNN, including single input, 10 hidden and 1 output layers, is used as a multi-class classifier. The input nodes in the input layer correspond to the number of attributes extracted from the dataset and are being optimized by the CS algorithm, whereas the number of nodes in the hidden layer corresponds to the feedback taken from the output layer to obtain the desired output. The output layer of the FFBPNN structure consists of a single neuron, which corresponds to the single output. The value '1' corresponds to normal data, and the value '0' indicates the presence of intrusion in the network. The designed model mainly includes four steps: upload data, optimization using CS, training, and data validation using FFBPNN. The general architecture of the work is shown in Fig. 3.

In the Fig. 4, features are selected based on the fitness function of the cuckoo search algorithm. The data that satisfies the fitness function is selected as the best feature of the data.

For any classification and prediction model, uniqueness and relevant feature selection is an essential step and compulsory to achieve better accuracy during classification or prediction. So, here cuckoo search algorithm is used to select the best set of features using their fitness according to the system requirements. For selecting a set of required and relevant feature, cuckoo search algorithm need a condition in terms of fitness that should be satisfied for selection of a feature as a set of optimized features.

A. Dataset

The CICIDS2017 data set includes benign and the latest common attacks, similar to real-world data (PCAP). The dataset is also composed of using CIC Flow Meter for network traffic analysis that is being labeled based on timestamp, the source IP, the destination IP, the source and destination address of port, protocol, and attack in the form of (CSV file). The extracted feature definition of the dataset is also available.

B. Upload Data

The foremost step in the proposed work is to upload data from the available dataset, as discussed above. After uploading the dataset, pre-processing has been performed to remove the undesired data.

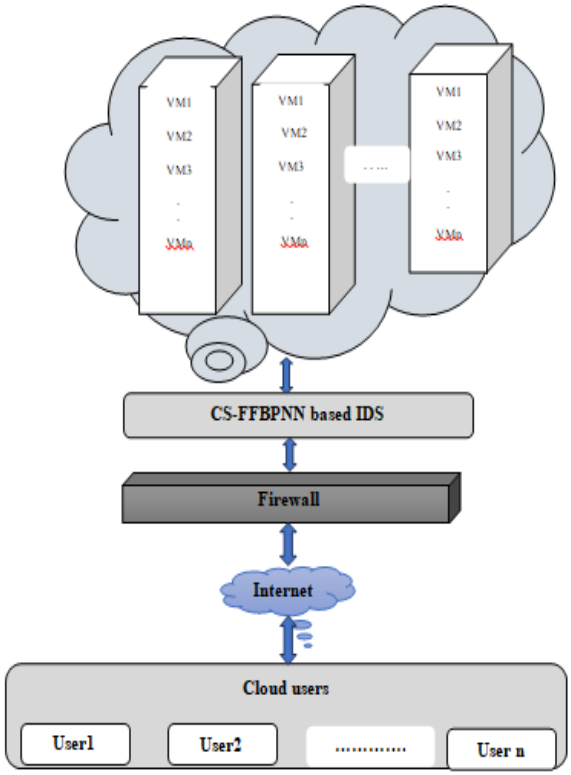

Fig. 3. General Workflow [25].

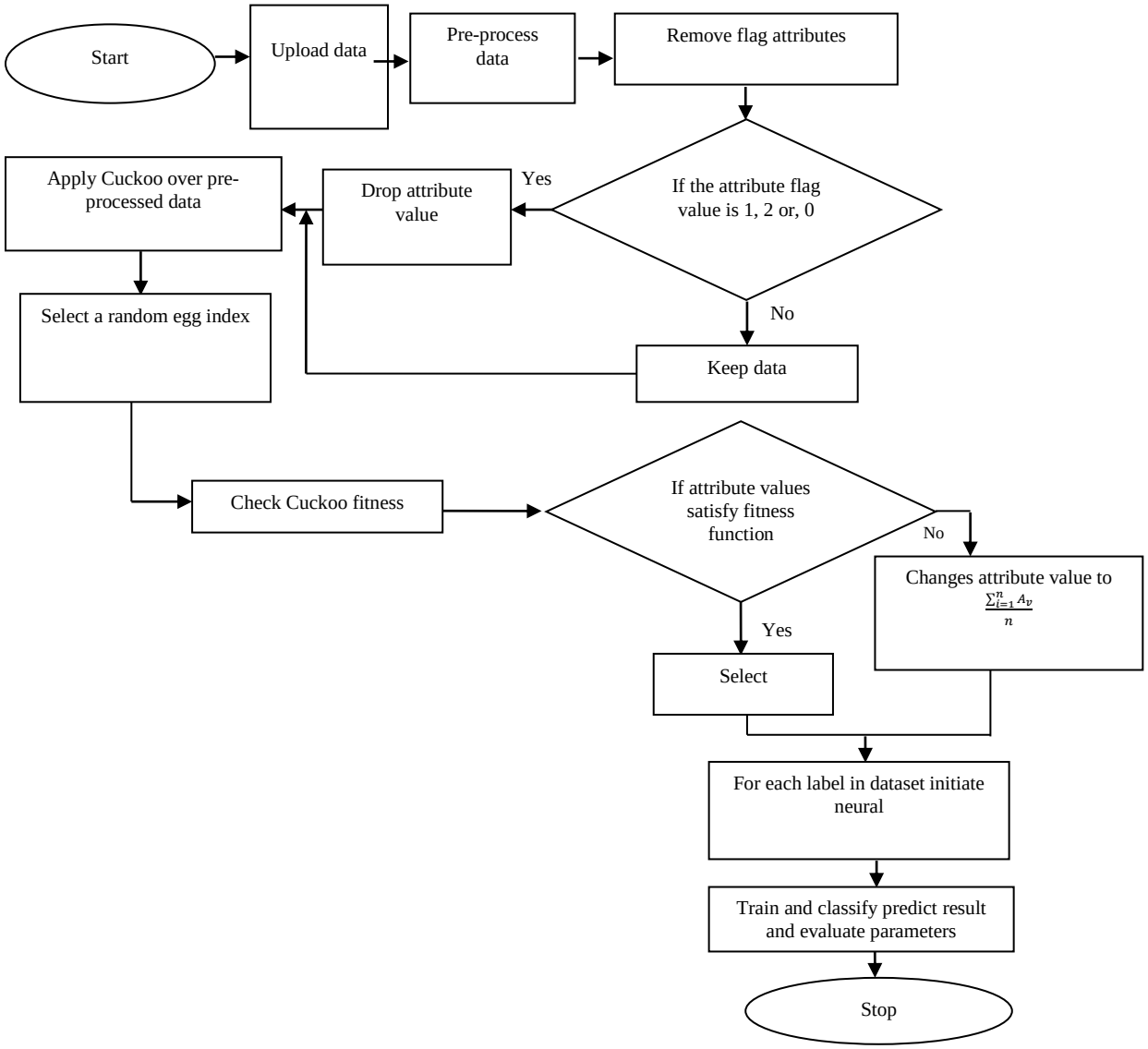

Fig. 4. Proposed Work Flow.

C. Pre-processing

Large disk space is required to store such records coming from multiple users in the cloud network. Therefore, to analyze each record in the log file, pre-processing, or the filtering of the desired record from the unwanted record is an essential step in data processing. It helps researchers to reduce computation time as well as saving the available resources. In other words, one can say that data Pre-processing is the way by which the raw data is processed for further processing. It converts complex/ unstructured data into a structured data form [26]. Here, pre-processing has been applied to remove the flag attributes. The case, when the value of flag attribute is 0, 1, and 2, then dropped that attribute value. Otherwise, passed data as input to the Cuckoo Search (CS) as an optimization algorithm.

Feature extraction plays an essential role in the IDS system. In feature extraction, mainly two processes are being carried out (i) feature construction and (ii) feature selection. The quality of both feature construction and feature selection is essential as they affect the classification accuracy of IDS. The feature construction, as well as the feature selection process, can be carried out manually or automatically using domain knowledge and machine learning approaches, respectively [27].

D. Cuckoo Search (CS) as an Attribute Optimizing Approach

CS is a metaheuristic approach used to optimize data based on the selected fitness function. The fitness function chosen is given in equation (5).

$$Ftness\ function = \begin{Bmatrix} S_b(true) & if S_b > Th_b \\ Th_b\ (false) & Otherwise \end{Bmatrix} \quad (5)$$

Here, S_b is the selected behavior of nodes, and Th_b denotes the threshold behavior of node, which is measured based on the average values of eggs generated by the cuckoos [28].

Based on the above-mentioned fitness criteria, a sensor node is considered as an intruder, if node requires more

energy or transmission time (delay) to forward or receive a data packet, otherwise considered as a normal node. After the segregation of all sensor nodes in the network in two categories, like normal and intruders, here FFBPNN is used to train the network of further classification of the intruders in the network and helps to prevent the network from different kinds of intruders. The working of which is explained in the subsequent section.

E. Feed Forward Back Propagation Neural Network (FFBPNN) as a Classifier

To prepare FFBPNN for detection of intruder automatically, the selection of the number of input neurons, hidden layer neurons must be decided accurately. To obtain the best performance, the modification of hidden neurons must be adequately done by following error detection and correction procedure, i.e., the generated error at the output layer is fed back to the hidden layer, which is again used for the modification of neurons in the hidden layer and then updates the weight matrix. The tangent function is used as an activation function in the hidden layer, whereas for output, the desired output is monitored using the linear activation function. After obtaining the desired output that is '1' for normal and '0' is for an intruder, it is then stored into the FFBPNN as training data. After selecting the appropriate structure, the designed IDS model for the cloud is trained based on the relationship between the input fed to the FFBPNN and the obtained output [13].

Where k is the constant of proportionality, 'error' is the error function, and w_{ij} denotes the weight between neuron i and j, respectively. The process of adjusting the weight matrix is repeated until the desired results are obtained, or the difference between the actual value and the node output is minimum as per acceptance level [13, 24]. The training scenario with a total 20 number of attributes and 30 hidden layers is shown on the right-hand side of Fig. 5.

The trained structure of FFBPNN with the generated Mean Square Error (MSE) is shown in Fig. 5. The network is trained at the 4th iteration with the minimum error of 1.2925 e^{-26} that is acceptance error, and after this, the network is trained, and the data is stored in its database.

After that the training testing process is performed to check the efficiency of the designed model, which is explained in the next section.

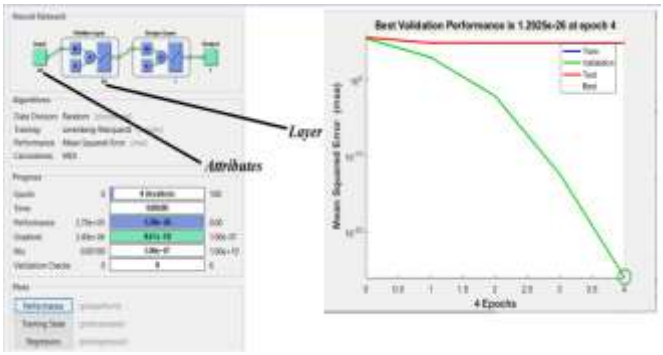

Fig. 5. Trained Structure of FFBPNN.

Algorithm: Enhanced CS_FFBPNN

Input: Pre-processed data (P-Data)
Output: Predicted Results (PR)
Initialization of variables
P-Data: Pre-processed data after the dataset loading
O-Data: Optimized Data as selected attributes from the dataset
Start selection
To optimized the P-data, CS Algorithm is used
Set up basic parameters of CS: Egg population in the nest ($E_{P\text{-}Data}$) = Number of P-Data
Define Fitness function for the selection best P-Data,

Fitness Function: $F(f) = \begin{cases} True; & if\ Selected_{P-Data} < Threshold_{P-Data} \\ False; & Otherwise \end{cases}$

In the fitness function, $selected_{-Data}$ is pre-processed current data present in the P-Data and $Threshold_{P-Data}$ is the threshold properties and the average of P-Data
Calculate Length of P-Data in terms of Row and Column, [Roz, Column] = Size (P-Data)
Set,O-Data= [] // Initiate an empty variable to store selected data
For i in rang of Row
For j in rang of Column
C_E = P-Data (i, j) = E_{P-Data} // Current egg from $E_{P\text{-}Data}$
$M_E = Threshold_{P-Data} = \sum_{i=1}^{Row} \sum_{j=1}^{Column} E_{P-Data}(i,j)$ // Mean of all $E_{P\text{-}Data}$
$F(f) = Fit\ Fun\ (C_E, M_E)$
O-Data = Cuckoo Search (F (f), FR (i, j))
End – For
End – For
Check the index of O-Data
If O-Data (index) = True
O-Data = Select (P-Data)
Else
O-Data = Reject (P-Data)
End –If
Returns: O-Data as a selected data
Initialize FFBPNN
Initialization of variables
→O-Data: Optimized Data as selected attributes from the dataset
→Cat: Target or Category according to the O-Data class
→N: Carrier Neurons Number
→PR: Predicted Results
Start training
Initialize FFBPNN- with O-Data: – Number of Epochs (E) // Iterations used by FFBPNN
– Number of Neurons (N) // Used as a carrier
– Performance: MSE, Gradient, Error Histogram, and Validation
– Data Division: Random
For i in range of all O-Data
If O-Data belongs to Type 1
Cat (1) = Feature from the O-Data of 1st Part // 1st Class of Dataset
Else (Others)
Cat (2) = Feature from the O-Data of 2nd Part // 2nd Class of Dataset
End – If
End – For
Initialized the pattern net using O-Data and Cat
FFBPNN--Structure = FFBPNN- (N)
Set the training parameters according to the requirements and train the system
FFBPNN-Structure = Train (FFBPNN-Structure, O-Data, Cat)
Test Data Group = Sim (FFBPNN-Structure, Current Data for testing)
If Test Data Group = 1
Predicted Results, PR = 1st with performance parameters
Else
Predicted Results, PR = 2nd with performance parameters
End – If
Return: FFBPNN--Structure as a trained structure with PR as a Predicted Results
End – Function
End – Function

V. Experimental Evaluation

Initially, the setup used to perform the analysis of the designed CS-FFBPNN based IDS for the cloud network is presented. Then the evaluated parameters are explained before and after the detection of an intruder in the cloud network.

A. Experimental Setup

The work is designed and implemented in MATLAB 2016 a. The evaluation was performed using an Intel core processor with 4 GB RAM. The designed IDS model's performance has been performed on two datasets, namely DoS attacks and BENIGN. The description of those datasets is provided in Table I.

TABLE I. Dataset Description

Category	Training data	Testing data
DoS attacks	241671	121451
BENIGN	92000	70000
Total	347241	199591

B. Experimental Result

The performance of the proposed research is evaluated in terms of different computation parameters, such as Classification Accuracy, Precision, Recall, and F-measure. All parameters are defined below.

1) Accuracy: It indicates the closeness of the detected output to the actual performance. Mathematically, it is given by equation (6).

$$Accuracy = \frac{T_p+T_N}{T_p+T_N+F_p+F_N} \quad (6)$$

where T_p → True Positive

T_N → True Negative

F_p → False Positive

F_N → False Negative

2) Precision: It represents the number of positively detected classes (normal or intruder) that belong to the positive class.

$$Precision = \frac{T_p}{T_p+F_p} \quad (7)$$

3) Recall: It represents the number of positive class (either normal or intruder) predictions made out of all positive samples saved in the FFBPNN database.

$$Recall = \frac{T_p}{T_p+F_N} \quad (8)$$

4) F-Measure: It represents a single score value that indicates the balance between the precision and recall parameter.

$$F\_measure = 2\times \frac{Precision*Recall}{Precision+Recall} \quad (9)$$

The comparison of precision parameters examined by the proposed work and existing work [20] with respect to a number of records is presented in Fig. 6. The X-axis represents the number of records, and the y-axis represents the precision values for proposed and existing work. The evaluated metrics values are summarized in Table II. Fig. 6 shows that with the increase in the number of records, the precision value also increases. This is due to the appropriate selection of data attributes so that the presence of an intruder is detected at the early stage and hence should be sorted before it affects the data transmission in the cloud network. From the graph, it is seen that precision for the proposed work is higher than existing work performed by Velliangiri *et al.* [20].

Proposed and the existing work for precision is 0.855and 0.775, respectively. Therefore, the percentage increase in precision from the existing work is 10.32 %.

The comparison of recall factor represents the TP entities corresponding to FN entities that are not at all categorized. In Fig. 7, the highest recall value for the proposed as well as for the existing work is recognized at 70,000 records, whereas the minimum recall value is analyzed at 10,000 records. The overall recall for multiple cloud users is analyzed as 0.8648. Also, the average value analyzed for the recall metric of the proposed work and existing work (TEHO-DBN) (Velliangiri et al., 2020) is 0.8648 and 0.812, respectively. Thus, the improvement in the recall values of 6.5% has been achieved against the existing work.

The values of F-measure after the detection of an intruder in the cloud network, for proposed as well as for the existing work for the same dataset, are shown in Fig. 8. The average F-score analyzed for proposed and existing work are 0.859, 0.790 respectively. Therefore, there is an improvement of 8.73% against the existing work.

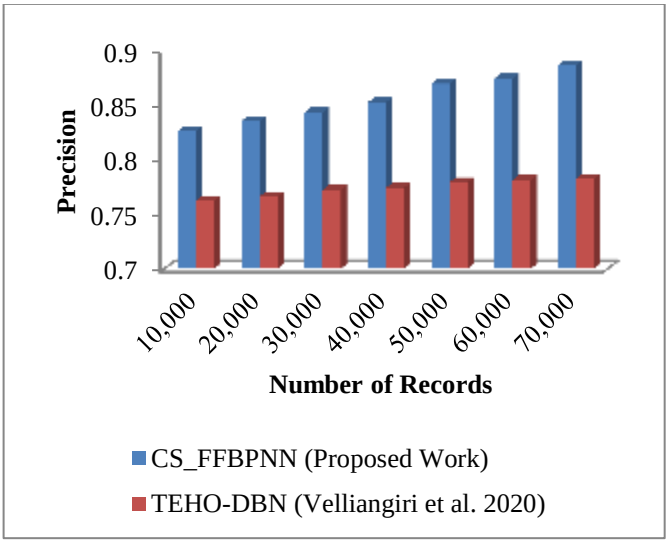

Fig. 6. Precision.

TABLE II. COMPUTED PARAMETRIC ANALYSIS

	Precision		Recall		F-measure		Classification Accuracy	
Number of Records	**CS_FFBPNN (Proposed Work)**	**TEHO-DBN (Velliangiri *et al.*, 2020)**	**CS_FFBPNN (Proposed Work)**	**TEHO-DBN (Velliangiri *et al.*, 2020)**	**CS_FFBPNN (Proposed Work)**	**TEHO-DBN (Velliangiri *et al.*, 2020)**	**CS_FFBPNN (Proposed Work)**	**TEHO-DBN (Velliangiri *et al.*, 2020)**
10,000	0.826	0.762	0.831	0.795	0.828	0.779	0.826	0.725
20,000	0.835	0.766	0.847	0.797	0.840	0.789	0.835	0.729
30,000	0.843	0.772	0.859	0.802	0.852	0.788	0.857	0.734
40,000	0.852	0.774	0.868	0.808	0.859	0.792	0.862	0.736
50,000	0.869	0.779	0.874	0.814	0.871	0.80	0.878	0.742
60,000	0.874	0.781	0.884	0.816	0.878	0.812	0.882	0.745
70,000	0.886	0.782	0.891	0.824	0.888	0.825	0.896	0.748

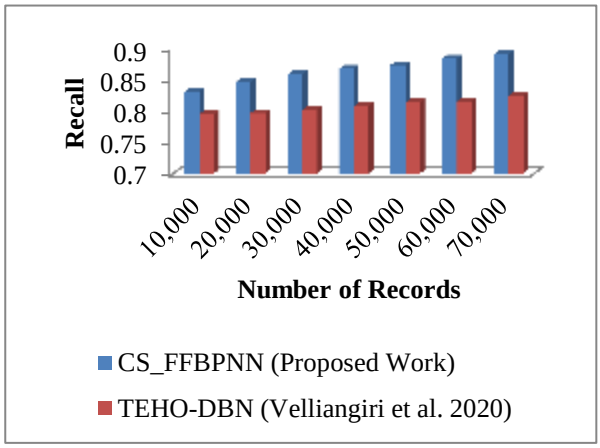

Fig. 7. Recall.

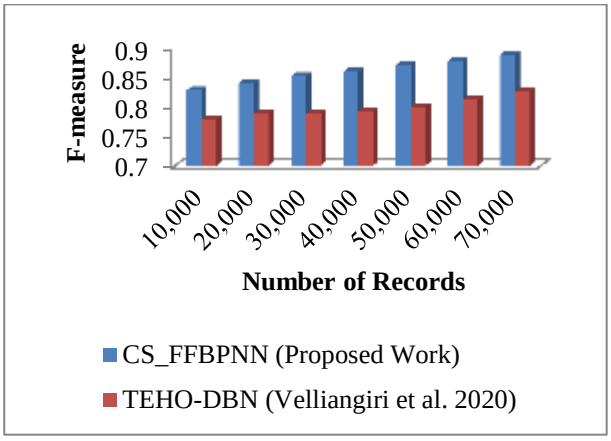

Fig. 8. F-measure.

To categorize the attacks considered in the database efficiently, IDS is used to ensure the security of the cloud data. To achieve better accuracy, proper training of the classification algorithm (FFBPNN) in this research is required. As the presence of irrelevant features in the dataset leads to increase in computation time, increase in error, and decrease classification accuracy. To solve this problem, the dataset is pre-processed and then optimized using a well-known Swarm Intelligence Cuckoo Search Approach (SI-CS) approach. The obtained results for the proposed IDS system for classification accuracy compared to existing work are shown in Fig. 9. Using CS-FFBPNN as IDS for the cloud system, the classification accuracy increases successfully. The graph shows improvement in the proposed work compared to the (Velliangiri *et al.*, 2020) work. The average classification accuracy analyzed for the proposed work, and the existing work are 0.8622 and 0.734. Therefore, the percentage increase in the classification rate from existing work is 17.47%.

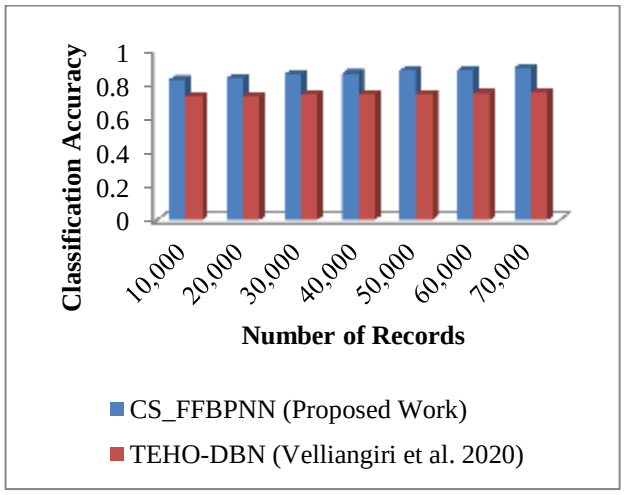

Fig. 9. Classification Accuracy.

The existing work represented by Velliangiri *et al.* (2020) is only suitable for DDoS attacks, but the proposed work is implemented on two different types of attacks which are DDoS and BENIGN. The proposed work provides better results as compared to existing approach. The proposed work also obtained the better values as compared to the existing approach against different parametric values, namely, precision, recall, classification accuracy, and F-measure.

VI. CONCLUSION

Cloud computing is used as a shared pool of resources that provides fast computing and aims to give convenient and, at the same time, required network access with minimal effort. The machine learning approach has offered the advantage of being interested in computing needs, and there is a suggestion for optimizing the unstructured data using the CS algorithm has also been presented. CS selected the optimal features from the pre-processed data and contributed to enhance the classification accuracy of the training and the testing phase of

the designed IDS cloud network. The proposed approach performed well in contrast to the existing work. As in existing work, authors have used a deep learning-based classifier in addition to Taylor-Elephant Herd Optimization, and the system time complexity increases as well as provides inadequate response for large network traffic. The work is evaluated against four performance metrics namely, precision, recall, f-measure, and classification accuracy of the designed IDS. From the test results, the examined accuracy has been observed as 86.22%. This accuracy might be low because the research has focused on detecting two types of attacks, such as DDoS attacks, and BENIGN attacks, for a sample of extensive data to train and test the network for intruder detection.

The proposed work is not implemented on real world data set. In future work, real world data set will be implemented by applying metaheuristic algorithm to achieve more accuracy based upon different types of attacks.

REFERENCES

[1] D.,Callegari, E., Conte, T., Ferreto, D., Fernandes, F., Moraes, F., Burmeister, & R. Severino. EpiCare—"A home care platform based on mobile cloud computing to assist epilepsy diagnosis", In 4th International Conference on Wireless Mobile Communication and Healthcare-Transforming Healthcare Through Innovations in Mobile and Wireless Technologies (MOBIHEALTH) , IEEE , pp. 148-151, November 2014.

[2] P., Mell, & T. Grance. "The NIST definition of cloud computing", 2011.

[3] M., Moorthy, & M, Rajeswari. "Virtual host based intrusion detection system for cloud. International Journal of Engineering & Technology", pp. 0975-4024, 2013.

[4] A., Carlin, M., Hammoudeh, & O., Aldabbas. "Defence for distributed denial of service attacks in cloud computing. Procedia computer science", vol. 73, pp. 490-497, 2015.

[5] S., Goyal. "Public vs private vs hybrid vs community-cloud computing: a critical review", International Journal of Computer Network and Information Security, vol. 6, no. 3, 2016.

[6] P., Ghosh, S., Shakti, & S, .Phadikar. "A cloud intrusion detection system using novel PRFCM clustering and KNN based dempster-shafer rule", International Journal of Cloud Applications and Computing" (IJCAC), vol. 6, no. 4, pp. 18-35, 2016.

[7] K., Indira, D., UshaNandini, & A., Sivasangari. "An efficient hybrid intrusion detection system for wireless sensor networks. Int J Pure Appl Math", vol. 119, no. 7, pp. 539-556 2018.

[8] M. A., Kumbhare & M. M., Chaudhari. IDS: "survey on intrusion detection system in cloud computing. Int. J. Comput. Sci. Mob. Comput"., vol. 3, no.4, pp. 497-502, 2014.

[9] E., Besharati, M., Naderan, & E., Namjoo, "LR-HIDS: logistic regression host-based intrusion detection system for cloud environments. Journal of Ambient Intelligence and Humanized Computing, vol. 10, no.9, pp. 3669-3692, 2019.

[10] T., Nathiya, & G., Suseendran. "An effective hybrid intrusion detection system for use in security monitoring in the virtual network layer of cloud computing technology". In Data management, analytics and innovation, Springer, Singapore, pp. 483-497, 2019.

[11] K., Hashizume, D., Rosado, G., Fernández-Medina, E., & E. B., Fernandez. "An analysis of security issues for cloud computing. Journal of internet services and applications", vol.4, no., pp. 1-13, 2013.

[12] P., Ghosh, A. K., Mandal, & R., Kumar. "An efficient cloud network intrusion detection system. In Information systems design and intelligent applications, Springer, New Delhi, pp. 91-99, 2015.

[13] N., Pandeeswari, & G., Kumar, "An anomaly detection system in a cloud environment using a fuzzy clustering-based ANN. Mobile Networks and Applications," vol. 21, no.3, pp. 494-505, 2016.

[14] M. M., Baig, M. M., Awais & E. S. M., El-Alfy. "A multiclass cascade of artificial neural network for network intrusion detection. Journal of Intelligent & Fuzzy Systems", vol. 32, no.4, pp. 2875-2883, 2017.

[15] V., Mahajan & S. K. Peddoju, "Deployment of the intrusion detection system in the cloud: a performance-based study", In IEEE Trust com/Big Data SE/ICESS, pp. 1103-1108. IEEE, August 2017.

[16] P., Deshpande, S. C., Sharma, S. K., Peddoju, & S. Junaid, HIDS: "A host-based intrusion detection system for a cloud computing environment", International Journal of System Assurance Engineering and Management, vol. 9, no. 3, pp. 567-576, 2018.

[17] B., Hajimirzaei, & N. J. Navimipour, "Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm", ICT Express, vol. 5, no. 1, pp. 56-59. 2019.

[18] K. R., Kumari, P., Sengottuvelan, & J. Shanthini, "A hybrid approach of genetic algorithm and multi-objective PSO task scheduling in cloud computing.", Asian Journal of Research in Social Sciences and Humanities, vol. 7, no. 3, pp. 1260-1271, 2017.

[19] M., Manickam, & S. P. Rajagopalan, "A hybrid multi-layer intrusion detection system in the cloud", Cluster Computing, vol. 22, no. 2, pp. 3961-3969, 2019.

[20] S., Velliangir, P., Karthikeyan & V. Vinoth Kumar, "Detection of distributed denial of service attack in cloud computing using the optimization-based deep networks", Journal of Experimental & Theoretical Artificial Intelligence, pp. 1-20, 2020.

[21] X. S., Yang, & S. Deb, "Cuckoo search via Lévy flights. In 2009 World congress on nature & biologically inspired computing", (NaBIC) pp. 210-214, IEEE, December, 2009.

[22] Jr, I., Fister Fister, D., & I. Fister, "A comprehensive review of cuckoo search: variants and hybrids", International Journal of Mathematical Modelling and Numerical Optimisation, vol. 4, no. 4, pp. 387-409, 2013.

[23] I., Fister, X. S., Yang, & D. Fister, "Cuckoo search: a brief literature review. In Cuckoo search and firefly algorithm", Springer, Cham, pp. 49-62, 2014.

[24] S. M., Mehibs & S. H. Hashim "Proposed network intrusion detection system in a cloud environment based on backpropagation neural network", Journal of the University of Babylon for Pure and Applied Sciences, vol. 26, no. 1, pp. 29-40, 2018.

[25] D. A. A. G., Singh, R., Priyadharshini, & E. J., Leavline "Cuckoo optimisation-based intrusion detection system for cloud computing." International Journal of Computer Network and Information Security, vol. 9, no.11, pp. 42-49, 2018.

[26] N., Paulauskas, & J., Auskalnis, "Analysis of data pre-processing influence on intrusion detection using NSL-KDD dataset. Open conference of electrical, electronic and information sciences (eStream), IEEE, pp. 1-5, 2017.

[27] G., Serpen, & E., Aghaei, "Host-based misuse intrusion detection using PCA feature extraction and kNN classification algorithms. Intelligent Data Analysis", vol. 22, no.5, pp. 1101-1114, 2018.

[28] V., Ravindranath, S., Ramasamy, R., Somula, K. S, Sahoo, & A. H, Gandomi, "Swarm intelligence-based feature selection for intrusion and detection system in cloud infrastructure." In IEEE Congress on Evolutionary Computation (CEC), pp.1-6, 2020.