Received on Nov 2020

Journal of Soft Computing and Engineering Applications

https://srkpublication.org/

Cyber Security Attack Detection Methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks in Cloud Computing

S. Giriprasad[1*], N. Kanthimathi[2]

[1]*Associate Professor, Department of Electronics and Communication Engineering, Coimbatore Institute of Engineering and Technology, Narasipuram, Tamil Nadu, India*

[2] *Assistant Professor, Electronics and Communication Engineering Department, Bannari Amman Institute of Technology, Sathyamangalam, Tamil Nadu, India*

[*]Corresponding author email: too.giri@gmail.com

Abstract: Cloud computing is an on-demand resource like data storage and computing power without direct supervision by the user. Even though the cloud computing has many advantages, it also have drawback, the main concern in a cloud environment is the organization of the cloud services. The possessions are allocated to the userthrough the cloud storage and there exists the vulnerabilities of cyber security attacks. Cyber security ensures practice of defending server, networks and data from malicious attacks. The classification and learning methods are utilized to enhance the cyber security against different attacks in different machine learning (ML) approaches. In this manuscript, a Cyber Security attack (CSA) detection methodology depending on a Chaotic Coyote Optimize Recurrent Neural Networks (CCO-RNN) in Cloud Computing is proposed. Here, the RNN model performs well on NSL-KDD dataset. The efficiency of proposed CSA-CCO-RNN system is compared with two existing methods, CSA-MOA-PSO and CSA-GA-SVM. The proposed CSA-CCO-RNN shows the node delay of 56.92% lower than CSA-MOA-PSO and 41.17% lower than CSA-GA-SVM and the proposed CSA-CCO-RNN shows the attack delay of 22.79% lower than CSA-MOA-PSO and 25.90% lower than CSA-GA-SVM algorithm. The experimental outcomes demonstrate that the CSA-CCO-RNN model is more efficient compared to existing methods.

Keywords: *Cloud computing, Cyber security, Chaotic Coyote Optimize, Recurrent Neural Networks, NSL-KDD dataset .*

1. INTRODUCTION

In current environment, wireless sensor networks (WSN) are extensively utilized in defence military, industrial facilities, aerospace, medical, health care, environmental monitoring, and further areas [1-3]. Because the size is small, resource-constrained sensor nodes (SN), monitoring circulation, dynamic change routing, how to efficiently address the difficulty of sensor network security is major concept of WSN tenders [4, 5]. In many of the real time applications wireless sensor networks are used, such as climate change and environmental watching, security and healthcare watching, and military investigation schemes [6-

8]. A WSN is a self-configured and infrastructure-less wireless networks. [9, 10] Wireless sensor network could be a portion of Internet of Things (IoT) with number of sensor nodes (SNs) composed. These SNs have circulated over a broad range of various areas to gather essential information and transfer the information to a central node. The central node is more proficient node, also known as base station (BS) node or sink node [11, 12].

In this work, Recurrent Neural Networks in Cloud Computing is considered. Neural network considers on the enhancement of computer programs that can learn them to develop and modify while exposed to new data. Neural network techniques have ability to execute a system that can learn from data. After learning, it can be utilized to classify new incoming packets as intrusive including normal packets. Nevertheless, the existing classification models are generally employed in practical issues, there are certain issues at the process of utilization viz, inadequate effects, minimal classification accuracy, poor adaptive capacity. This motivated us to concentrate on modelling an effective attack classifier for cloud computing for cyber security attack.

The main contribution of this manuscript is summarized as follows:

- In this manuscript, an effective Cyber Security attack detection method for Chaotic Coyote Optimize Recurrent Neural Networks (CSA-CCO-RNN) is proposed for detecting the Cyber Security attacks in cloud computing.
- Here, the recurrent neural network (RNN) uses the Chaotic Coyote Optimization algorithm to optimize the function and learn the classification
- The performance of the proposed CSA-CCO-RNN is evaluated through rigorous computer simulation by using the publically available database i.e. NSL-KDD dataset.
- The efficiency of proposed Chaotic Coyote Optimize Recurrent Neural Networks (CCO-RNN) is compared with existing methods by considering the evaluation metrics.
- To demonstrate the efficiency of the proposed multi class model in attacks classification, the investigation on various dataset is carried out because every dataset affects from different problems viz, data corruptions, inconsistencies, traffic, out of date, contemporary attacks.

2. LITERATURE SURVEY

Several research works are already existed in literature based on cyber-attack detection and defence strategies. Some of the most recent works of are reviewed here.

In 2018, Ding et al [13] have presented a security control and attack detection of industrial cyber-physical systems (CPS). Here, the CPS plays vital role in critical infrastructure, administration, daily life through the incorporation of computation, networking, physical processing. An industrial CPSs attack detection was developed according to the categories on detection methods. Also we discussed about the safe control with state assessment.

In 2018, Sahoo et al [14] have suggested the Stealth Cyber-Attack Detection approach for direct current Micro grids. Here, the cooperative mechanism for detecting the illusory of cyber-attack in the average voltage controlling as well as current sharing cyber-physical direct current micro grids is described. Moreover, they discussed about the development and related scope of stealth attacks to mislead dispersed observers realizing the essential with adequate terms to model of specified attacks. Here, an innovative cooperative vulnerability factor (CVF) system for every agent was established, which determines the attacked agent accurately under different scenarios. The experimental result was validated to error data injection, stealth attacks in sensors, communication links.

In 2019, Zhang et al [15] have presented the multilayer data-driven cyber attack detection for Industrial Control Systems (ICS) in terms of network system. Here, the current ICS cyber security effort was depending on firewalls, data diodes, and other intrusion prevention models that cannot be adequate to increase cyber threats as moved attackers. The cyber attack was detected by monitoring the host system's network and data. The experimental result shows the presented approach was detect the physically impact full cyber attack before noteworthy effects occur.

In 2020, Poornima et.al [16] presented an online locally weighted projection regression (OLWPR) for recognition of anomalies at WSN. LWPR model was not parametric, also the current forecasts were activated in local operations using only the data subset. Therefore, the calculation complexity became less that was a request in WSN. After the forecast process, the dynamic threshold value was defined as dynamic threshold model for recognizing the digression of forecast value as actual detected value. OLWPR reaches a detection rate of 86% and an error rate well below 16%.

In 2017 Wang et.al in [17] have presented an innovative intrusion detection system (IDS) named hierarchical spatial-temporal features-based IDS (HAST-IDS), here initially the lower-level spatial characteristics of network traffic utilizing deep convolutional NNs. The overall procedure of characteristic learning was done using deep NNs automatically. The automatically learned traffic characteristics successfully diminish the false alarm rate. The standard datasets DARPA1998 and ISCX2012 were employed for assessing the system efficiency.

In 2018, Hartanto R et al. [18] have introduced Distributed Denial of Service (DDoS) attack detection depending on simple artificial NN with Synthetic Minority Over-sampling Technique (SMOTE) for internet of things Environment. Their detection system utilizes the public dataset for noticing attack using machine learning technique by modern botnet Bot-IoT .They used SMOTE for resolving disparity data issue to appliance a machine learning-based

distributed denial of service detection system. The disadvantage was it only concentrated on DDoS attack for IoT environment.

In 2018, Abusitta A et al. [19] have presented a framework based on SVM to detect DoS attacks on virtualized clouds under varying environments. Here, the presented work provides an SVM-based framework to detect DoS attack detection on virtualized cloud under varying environments. They use a filter to remove the noisy data in the preprocessing step. The presented work was better than traditional SVM classifier. The drawback was it uses a filter it reduces the detection accuracy.

In 2017, Zekri M et al. [20] have suggested that DDoS attack detection utilizing machine learning approaches on cloud computing environments including C.4.5 algorithm and decision tree. They calculated DDoS detection scheme activated on C.4.5 algorithm to alleviate the distributed denial of service threat. The suggested approach attached with signature detection approach and produces a decision tree to carry out programmed, operative discovery of signatures attacks for distributed DoS flooding attacks. It also deliberates about three practices of the Intrusion detection and establishes about the C4.5 model. The drawback of the suggested work was due to C4.5 algorithm can able to detect the DDoS attack only when it coupled with the signature detection techniques.

3. PROPOSED METHOD FOR CYBER SECURITY ATTACK DETECTION METHODOLOGY BASED ON A CHAOTIC COYOTE OPTIMIZE RECURRENT NEURAL NETWORKS IN CLOUD COMPUTING

In this section, Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks in Cloud Computing is proposed [21, 22]. In the cyber security attack detection methodology with the main function and how this system works are shown in Figure 1. A user sent the request into the system, that request is send to the attack detection block. This block contains 3 major operations i.e. (i) data collection with pre-processing, (ii) attack recognition (iii) request processing. The collection of data is accountable for data collection with pre-processing the request for recurrent neural network classifier. This operation is to improve the proficiency of the proposed method, also assists the CCO approach that is employed in the faster assemble to training model. (ii) Recognition of attack is used to categorize the received request depending on the trained recurrent neural network classifier mode. After the recurrent neural network classifier mode is trained in a disconnected mode, it is applied to identify malicious requests, (iii) Request processing, the attack detection function marks the incoming request either as typical or suspicious request. If the request is typical then it is functioned by available cloud resources. Then, the suspicious request is reported for the module of security control.

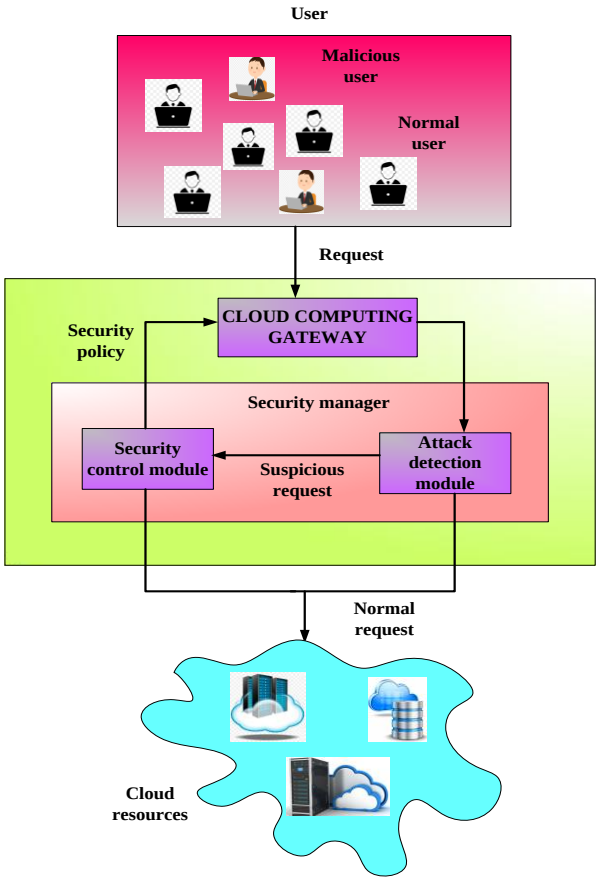

Figure 1: Block diagram for proposed Cyber Security attack detection methodology

In the security control unit, the suspicious request is activated by the request varying function. Here the suspicious request comparing and confirmed with the present dataset are send to the security services providers for cross check. Identify the request is to be safe it can help as the normal requests. In difference the request is preserved as a malicious request including attack secure the security operation is implement rapid security policies are activating to prevent the impacts of spread and its attack. For example, Google hacking attack is to identify the request; the security manager can directly implement to the security measures for web associated susceptibilities by implementing the backup policies to the data recovery avoidances for example Continuous Data Protection (CDP) in the level of application. This is considered as the basic

In recent years, world move towards cloud computing, so it become more complicated and attacking can happen at changed shares of the cloud like the data storage, during a transaction, during resource utilization and sharing. Some of the potential cyber security attacks on cloud computing are DoS attack. In this case Minkowski Distance is identified variance between the classes, based on this variance is determined as two points are,

$$q_1 = (y_1, y_2, y_3, \ldots, y_m); q_2 = (x_1, x_2, x_3, \ldots, x_m) \in C^m \quad (1)$$

$$\left(\sum_{j=1}^{m} | y_j - y_j \pm n | \right)^{lq} l^{\frac{1}{q}} \quad (2)$$

The variance or distance amid any 2 points is expressed in following equation,

$$u_{ji} = \left(\sum_{j=1}^{m} | y_{jk} - y_{ik} |^{lq} \right) l \frac{1}{q} \quad (3)$$

Where lq implies sequence or centroid of class. The given derivation signifies the centroid of the class,

$$C_j = \sum_{n=1}^{dt} J_n^j / m_j \quad (4)$$

Where, C_j implies the centroid value of j^{th} node, J denotes the individual j^{th} lower most distance or variance, and d represents the count of magnitudes. Then, the homogeneous models of tuple characteristics control for centroid of every mode class. After the determination, the classes are detected with various process of pattern. At DoS attack, an attacker overloads the besieged cloud system along service requests data packets persistently for besieged cloud server, without modifying the nodes of data packets, or decrypting encrypted data so that cloud system stops response to any upcoming requests, so the completed resources unavailable for users and it consumes more network bandwidth.

3.1 Step by step procedure for Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks

In this step Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks in cloud computing is discussed. Cyber security attack detection is to improve a fast and effective intelligent abnormality detection system to conflict growing attacks. The individuality of the CSA-CCO-RNN method is classically the loss functions with ReLU to make the most of deep learning efficiently. Chaotic Coyote Optimized Algorithm (CCOA) is separated as packs, then the internal social impact is estimated. Here, initialize the initial population size, to optimize the hyper parameters together with count of packs, population size, maximum count of generations and optimizes the single objective function. Step by step procedure or the Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks are given below,

Step 1: Initialization
Here initialize the initial population size A is composed with number of packs Q_m with the number of coyotes and then initialize the $(iteri = 0)$ within the search space and determined through the interval $|(qx, jx)|$ is obtained by the following equations

$$O_{h,z}^{l,s} = qx + t_z.(jx_z - qx_z) \quad (5)$$

$$h = [1,2,..A_h], l = [1,2,..A_l], z = [1,2,..E] \quad (6)$$

Where, E denotes dimension of optimization issue, t_z specifies random number within [0,1] determined uniform probability distribution (UPD). Here, packs have been randomly categorized using similar distribution, also initial coyote's ages $(D_h^{l,0})$ are equal to zero.

Step 2: Objective function
The objective function is defined as coyote's adaptation. This is the outcome of its social condition (SC).

$$W_h^{l,s} = w(O_h^{l,s}) \quad (7)$$

Step 3: Estimate the Fitness function
The sigma coyote represents the better SC. Here, CCOA means the best smallest or highest value for selecting best sentiment analysis i.e., positive or negative sentiments, then the objective function cost is given as

$$K\varsigma^{l,s} = \{O_h^{l,r} \mid \arg u_{h=\{1,2,..A_h\}} \min_{ej(O_h^{l,s})} \mid \} \quad (8)$$

Step 4: Social tendency
The coyote's social behaviour is natural influence through the sigma. By applying CCOA denotes the cultural pack tendency $(hs^{l,s})$, it represents median of coyote's SC.

$$(hs^{l,s}) = med(O_{h,z}^{l,s}) \forall_h \in \{1,2,..A_h\} for = [1,2,..E] \quad (9)$$

For each h^{th} coyote of the q^{th} pack do again steps 5 to 7

Step 5: Update the social condition (SC)
The SC updates by the impact of sigma coyotes ς_u and social trend ς_s, is created as two random coyotes of pack $(ht_1 and ht_2)$, it is given as

$$C\_O_h^{l,s} = O_h^{l,s} + t_1.\varsigma_s + t_2.\varsigma_u \quad (10)$$

$$\varsigma_s = hs^{l,s} - O_{ht_1}^{l,s} \quad (11)$$

$$\varsigma_u = K^{l,s} - O^{l,s}_{ht_1} \quad (12)$$

$t_1 and t_2$ are the social pack and the weight of the pack with the sigma influences with both random numbers within [0,1] range created to random with the uniform probability distributions.

Step 6: Evaluation
Cost intention function is determined by process of SC.

$$C.W^{l,s}_h = f(C\_O^{l,s}_h) \quad (13)$$

Step 7: Adaptation
Coyotes select that SC, in which the optimal fits the environment for maintaining the subsequent iteration, that is optimal small or high for selecting best sentiment analysis for positive or negative sentiments, and then the objective function cost is given as,

$$O^{l,s+1}_h = \begin{cases} c\_O^{l,s}_h, C\_W < W^{l,s}_h \\ O^{l,s}_h, otherwise \end{cases} \quad (14)$$

Step 9: Transition
Here the life of coyote can eliminate from the pack, then transferred to other. Here, 2 coyotes from various packs alter its locations along probability l_{qs} and given as

$$l_{qs} = 0.005.A^2_h \quad (15)$$

Step 10: Updating
In this the ages are updated each iterations and it is given as

$$D^{l,s+1}_h = D^{l,s}_h + 1 \quad (16)$$

Step 11: Solution Selection
To determine the optimization issue, the optimal adapted coyote between the whole packs are chosen from the solution.

Step 12: Termination
Chaotic Coyote optimization algorithm (CCOA) optimizes the objective function such as increase the accuracy in the cyber security attack detection. Here the features are extracted using the recurrent neural network (RNN) classifier. To get more accuracy Chaotic Coyote optimization algorithm (CCOA) is proposed.

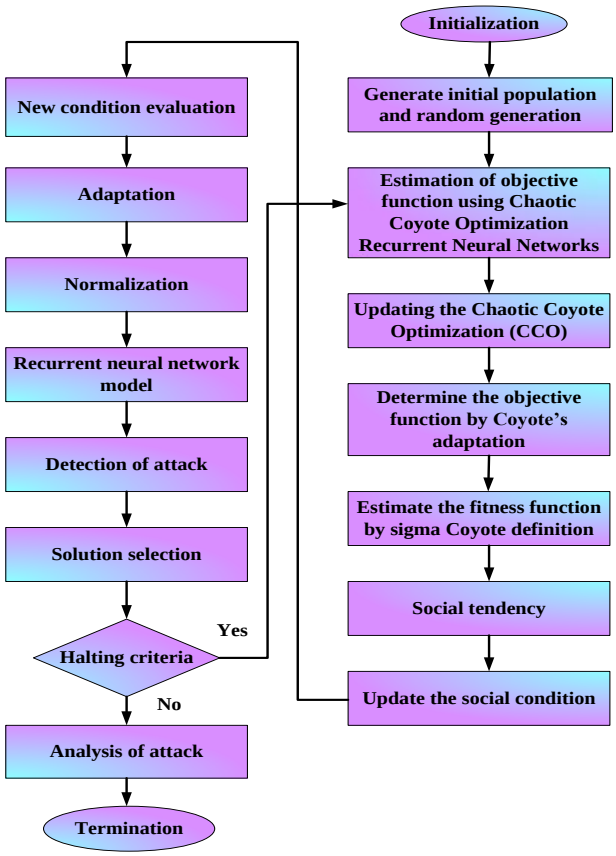

Figure 2: Flow chart for Cyber Security attack detection methodology based on a Chaotic Coyote Optimization Recurrent Neural Networks in Cloud Computing

4. RESULT AND DISCUSSION

To verify the performance of the Chaotic Coyote Optimization Recurrent Neural Networks (CCO -RNN) model for Multiclass Cyber-Attack Classification, this work use public dataset, that is. NSL-KDD dataset. To simulation track of the proposed algorithm, every calculations performs in MATLAB site on Windows 7 system along Intel (R) Core (TM) i7-4790 CPU @ 3.6 GHz through 8 GB of RAM. The simulations perform by deeming count of WSN scenarios, where the count of SNs varies from 300 to 700 and the count of gateways from 60 to 90. The performance of the proposed method Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks (CSA-CCO-RNN) is compared with existing methods such as Cyber Security attack detection based on magnetic optimization algorithm particle swarm optimization (CSA-MOA-PSO) [23] and Cyber Security attack detection based genetic algorithm support vector machine (CSA-GA-SVM) [24].

4.1 Network traffic-based dataset: NSL-KDD dataset

For authenticating the accuracy of Chaotic Coyote Optimization Recurrent Neural Networks model for cyber security attack detection using public datasets that is NSL-KDD dataset [25]. The NSL-KDD data set is applied for addressing the issues inherent in KDD'99 data set. NSLKDD dataset is improved by the following steps that is (i) redundant records are not involve (ii) duplicate records are not involve (iii) the count of choosen records is designed from % of records (iv) the number of records are responsible, the above improvements are likened to the original KDD dataset. In NSLKDD is typically found the best result in the intrusion detection system.

4.2 Simulation phase 1: performance comparison of various methods

Figure 3 to 7 shows the simulation result for Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks in Cloud Computing. The various performances like node Vs delay, node Vs delivery ratio, Node Vs Fairness Index, Node Vs Overhead, and Node Vs Throughput are analyzed.

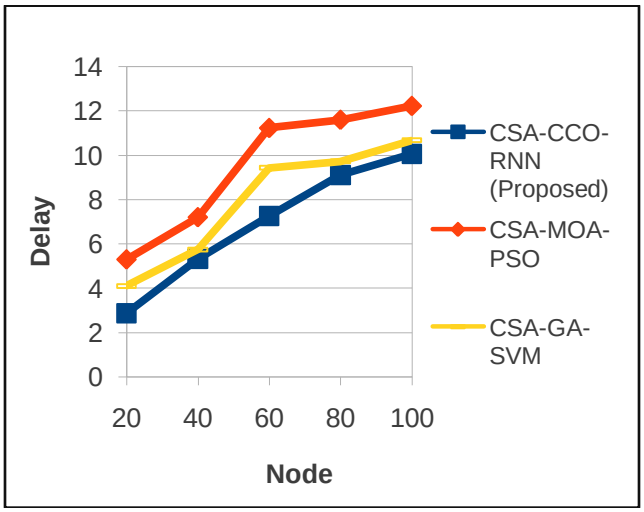

Figure 3: Node Vs delay

Figure 3 shows the node Vs delay in network scenario. The delay of the proposed method CSA-CCO-RNN is compared with existing method such as CSA-MOA-PSO and CSA-GA-SVM. At node 20, the proposed method produces 36.92% lower than CSA-MOA-PSO method and 41.17% lower than CSA-GA-SVM method respectively. At node 40, the proposed method produces 37.13% lower than CSA-MOA-PSO method and 8.45% lower than CSA-GA-SVM method. At node 60, the proposed method produces 46.46% lower than CSA-MOA-PSO method and 34.24% lower than CSA-GA-SVM method. At node 80, the proposed method produces 32.47% lower than CSA-MOA-PSO method and 6.87% lower than CSA-GA-SVM method. At node 100, the proposed method produces 7.17% lower than CSA-MOA-PSO method and 32.47% lower than CSA-GA-SVM method.

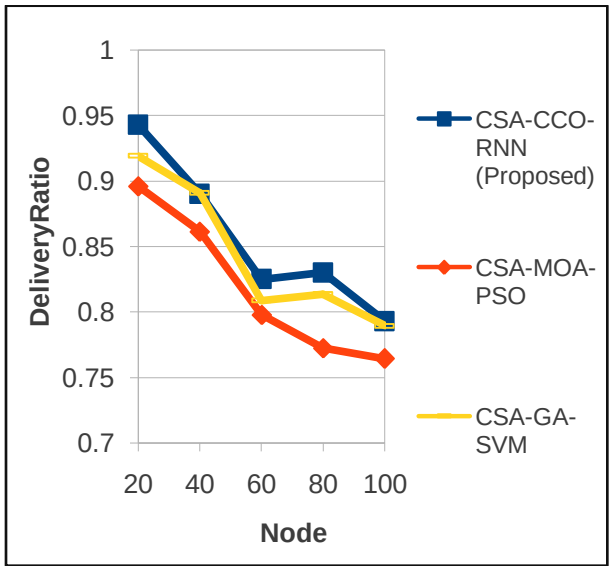

Figure 4: Node Vs delivery ratio

Figure 4 displays that node Vs delivery ratio on network scenario. The delivery ratio of the proposed method CSA-CCO-RNN is compared with existing method such as CSA-MOA-PSO and CSA-GA-SVM. At node 20, the proposed method produces 3.69% higher than CSA-MOA-PSO method and 6.35% higher than CSA-GA-SVM method respectively. At node 40, the proposed method produces 4.44% higher than CSA-MOA-PSO method and 1.18% higher than CSA-GA-SVM method. At node 60, the proposed method produces 4.49% higher than CSA-MOA-PSO method and 2.98% higher than CSA-GA-SVM method. At node 80, the proposed method produces 8.57% higher than CSA-MOA-PSO method and 3.12% higher than CSA-GA-SVM method. At node 100, the proposed method produces 4.85% higher than CSA-MOA-PSO method and 1.52% higher than CSA-GA-SVM method.

Figure 5 shows the node Vs fairness index in network scenario. The fairness index of the proposed method CSA-CCO-RNN is compared with existing method such as CSA-MOA-PSO and CSA-GA-SVM. At node 20, the proposed method produces 6.37% lower than CSA-MOA-PSO method and 37.64% lower than CSA-GA-SVM method respectively. At node 40, the proposed method produces 7.99% lower than CSA-MOA-PSO method and 29.10% lower than CSA-GA-SVM method. At node 60, the proposed method produces 39.98% lower than CSA-MOA-PSO method and 22.53% lower than CSA-GA-SVM method. At node 80, the proposed method produces 39.65% lower than CSA-MOA-PSO method and 21.97% lower than CSA-GA-SVM method. At node 100, the proposed method produces 31.42% lower than CSA-MOA-PSO method and 32.47% lower than CSA-GA-SVM method.

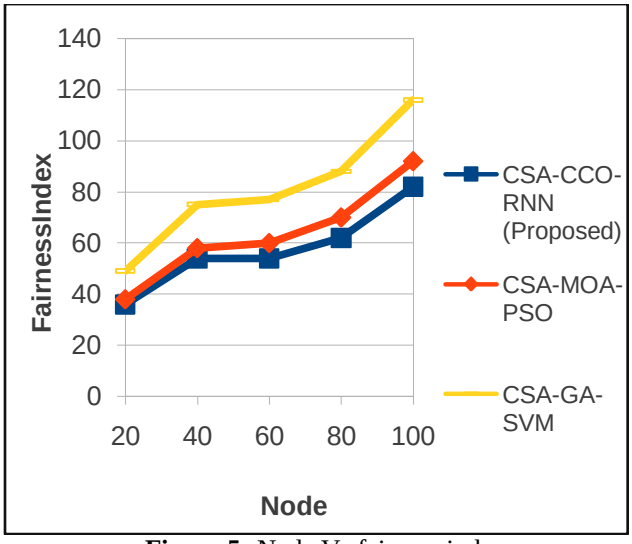

Figure 5: Node Vs fairness index

Figure 6 shows the node Vs overhead in network scenario. The overhead of the proposed method CSA-CCO-RNN is compared with existing method such as CSA-MOA-PSO and CSA-GA-SVM. At node 20, the proposed method produces 6.23% lower than CSA-MOA-PSO method and 28.21% lower than CSA-GA-SVM method respectively. At node 40, the proposed method produces 7.89% lower than CSA-MOA-PSO method and 28.21% lower than CSA-GA-SVM method. At node 60, the proposed method produces 21.32% lower than CSA-MOA-PSO method and 30.24% lower than CSA-GA-SVM method. At node 80, the proposed method produces 22.58% lower than CSA-MOA-PSO method and 30.63% lower than CSA-GA-SVM method. At node 100, the proposed method produces 20.93% lower than CSA-MOA-PSO method and 30.45% lower than CSA-GA-SVM method.

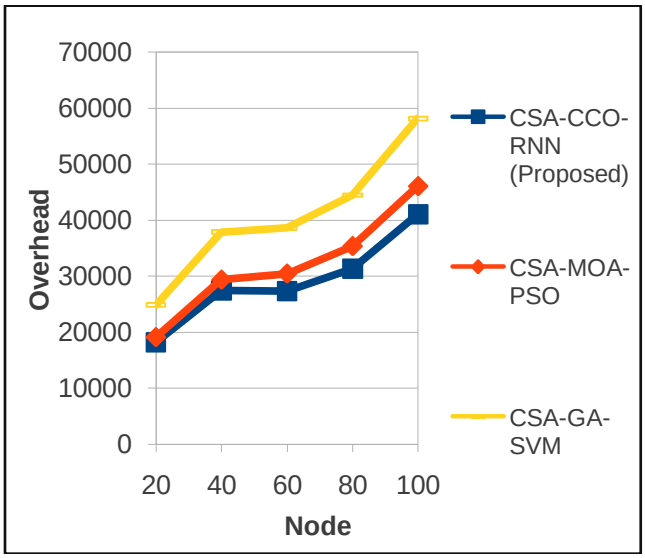

Figure 6: Node Vs overhead

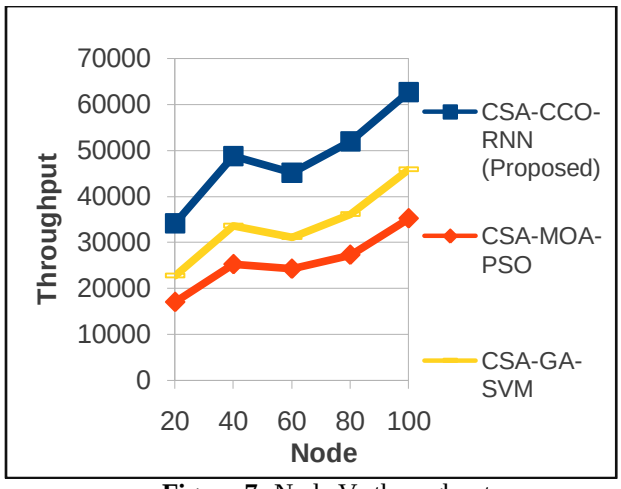

Figure 7: Node Vs throughput

Figure 7 shows the node Vs throughput in network scenario. The throughput of the proposed method CSA-CCO-RNN is compared with existing method such as CSA-MOA-PSO and CSA-GA-SVM. At node 20, the proposed method produces 99.79% higher than CSA-MOA-PSO method and 59.87% higher than CSA-GA-SVM method respectively. At node 40, the proposed method produces 93.76% higher than CSA-MOA-PSO method and 55.70% higher than CSA-GA-SVM method. At node 60, the proposed method produces 55.63% higher than CSA-MOA-PSO method and 96.77% higher than CSA-GA-SVM method. At node 80, the proposed method produces 91.35% higher than CSA-MOA-PSO method and 54.88% higher than CSA-GA-SVM method. At node 100, the proposed method produces 88.94% higher than CSA-MOA-PSO method and 47.63% higher than CSA-GA-SVM method.

4.3 Simulation phase 2: performance comparison of various algorithms

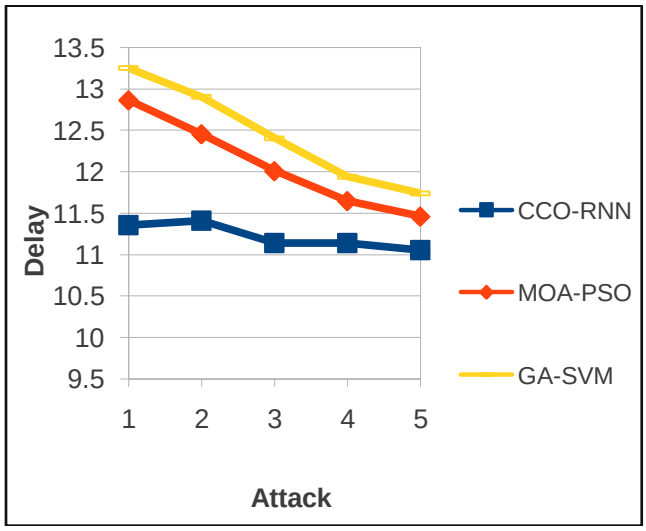

Figure 8: Attack Vs delay

Figures 8 to 12 shows the simulation result for Cyber Security attack detection methodology based on a Chaotic Coyote Optimize Recurrent Neural Networks in Cloud Computing. The various performances like attack Vs delay, attack Vs delivery ratio, attack Vs drop, attack Vs energy consumption, and attack Vs network lifetime are analyzed.

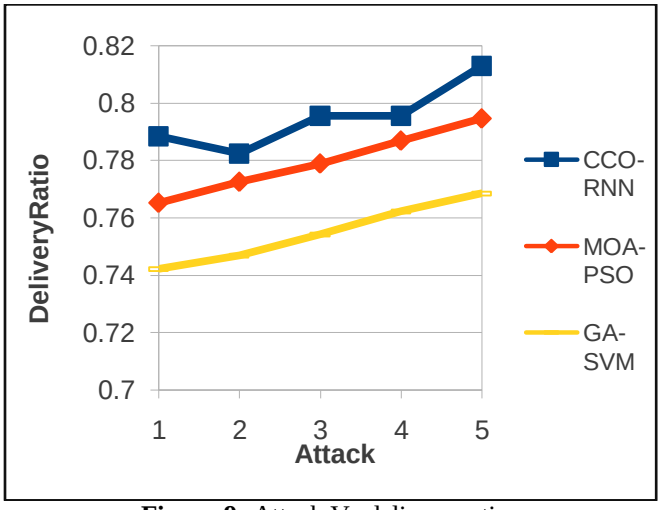

Figure 9: Attack Vs delivery ratio

Figure 8 shows the attack vs delay value for network scenario. The delay of proposed CCO-RNN algorithm is compared with two existing algorithm. The two existing algorithms are MOA-PSO and GA-SVM algorithms. At attack 1, the proposed CCO-RNN algorithm shows the attack delay of 11.68% lower than MOA-PSO algorithm and 14.29% lower than GA-SVM algorithm. At attack 2, the proposed CCO-RNN algorithm shows the attack delay of 8.36% lower than MOA-PSO algorithm and 11.57% lower than GA-SVM algorithm. At attack 3, the proposed CCO-RNN algorithm shows the attack delay of 7.22% lower than MOA-PSO algorithm and 10.18% lower than GA-SVM algorithm. At attack 4, the proposed CCO-RNN algorithm shows the attack delay of 4.36% lower than MOA-PSO algorithm and 6.72% lower than GA-SVM algorithm. At attack 5, the proposed CCO-RNN algorithm shows the attack delay of 3.52% lower than MOA-PSO algorithm and 5.82% lower than GA-SVM algorithm.

Figure 9 shows the attack vs delivery ratio value for network scenario. The delivery ratio of proposed CCO-RNNalgorithm is compared with two existing algorithms. The two existing methods are MOA-PSO and GA-SVM algorithms. At attack 1, the proposed CCO-RNNalgorithm shows the attack delivery ratio of 3.01% higher than MOA-PSO algorithm and 6.23% higher than GA-SVM algorithm. At attack 2, the proposed CCO-RNNalgorithm shows the attack delivery ratio of 1.27% higher than MOA-PSO algorithm and 4.76% higher than GA-SVM algorithm. At attack 3, the proposed CCO-RNNalgorithm shows the attack delivery ratio of 2.14% higher than MOA-PSO algorithm and 5.50% higher than GA-SVM algorithm. At attack 4, the proposed CCO-RNNalgorithm shows the attack delivery ratio of 1.11% higher than MOA-PSO algorithm and 4.38% higher than GA-SVM algorithm. At attack 5, the proposed CCO-RNNalgorithm shows the attack delivery ratio of 2.31% higher than MOA-PSO algorithm and 5.78% higher than GA-SVM algorithm.

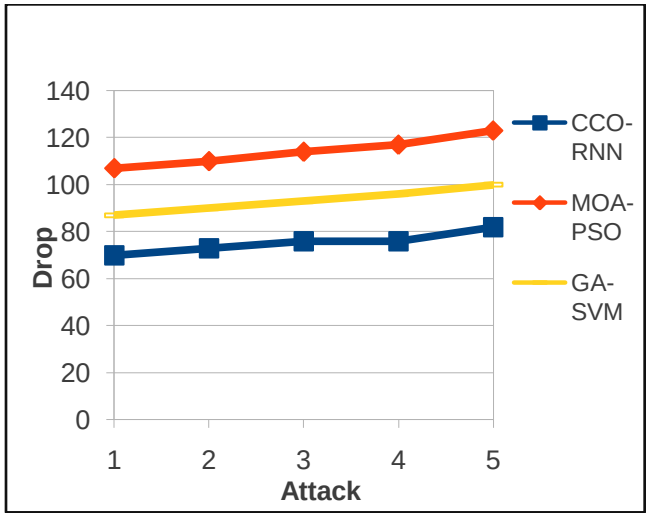

Figure 10: Attack Vs drop

Figure 10 shows the attack vs drop value for network scenario. The drop of proposed CCO-RNN method is compared with two existing algorithms. The two existing algorithm are MOA-PSO and GA-SVM algorithms. At attack 1, the proposed CCO-RNN algorithm shows the attack drop of 34.57% lower than MOA-PSO algorithm and 19.54% lower than GA-SVM algorithm. At attack 2, the proposed CCO-RNN algorithm shows the attack drop of 33.63% lower than MOA-PSO algorithm and 18.88% lower than GA-SVM algorithm. At attack 3, the proposed CCO-RNN algorithm shows the attack drop of 33.33% lower than MOA-PSO algorithm and 18.27% lower than GA-SVM algorithm. At attack 4, the proposed CCO-RNN algorithm shows the attack drop of 35.04% lower than MOA-PSO algorithm and 20.83% lower than GA-SVM algorithm. At attack 5, the proposed CCO-RNN algorithm shows the attack drop of 33.33% lower than MOA-PSO algorithm and 18% lower than GA-SVM algorithm.

Figure 11 shows the attack vs energy consumption value for network scenario. The energy consumption of proposed CCO-RNN algorithm is compared with two existing algorithm. The two existing methods are MOA-PSO and GA-SVM algorithms. At attack 1, the proposed CCO-RNN algorithm shows the attack energy consumption of 2.16% lower than MOA-PSO algorithm and 2.41% lower than GA-SVM algorithm. At attack 2, the proposed CCO-RNN algorithm shows the attack energy consumption of 1.27% lower than MOA-PSO algorithm and 2.74% lower than GA-SVM algorithm. At attack 3, the proposed CCO-

RNN algorithm shows the attack energy consumption of 3.67% lower than MOA-PSO algorithm and 4.42% lower than GA-SVM algorithm. At attack 4, the proposed CCO-RNN algorithm shows the attack energy consumption of 0.87% lower than MOA-PSO algorithm and 2.88% lower than GA-SVM algorithm. At attack 5, the proposed CCO-RNN algorithm shows the attack energy consumption of 0.76% lower than MOA-PSO algorithm and 1.52% lower than GA-SVM algorithm.

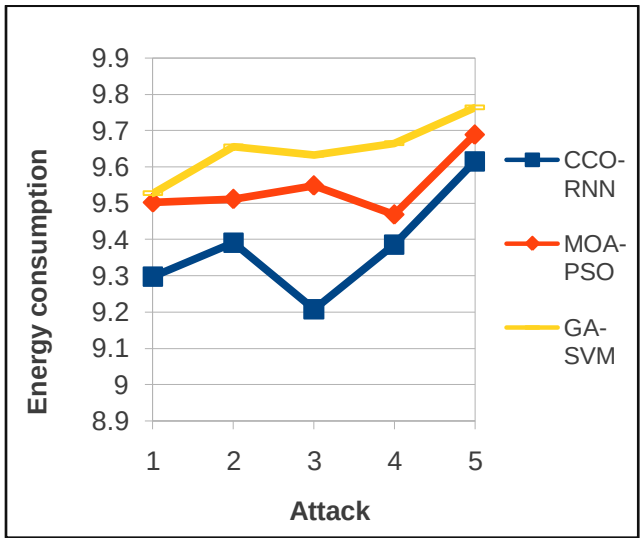

Figure 11: Attack Vs energy consumption

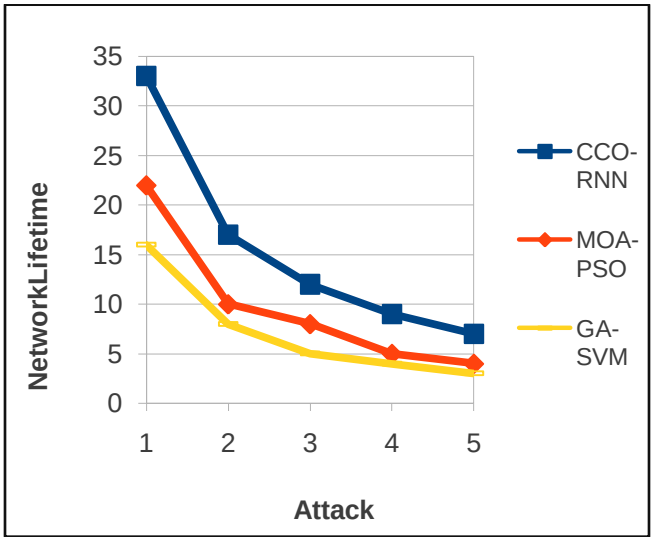

Figure 12: Attack Vs network lifetime

Figure 12 shows the attack vs network life time value for network scenario. The network life time of proposed CCO-RNN algorithm is compared with two existing algorithm. The two existing algorithms are MOA-PSO and GA-SVM algorithms. At attack 1, the proposed CCO-RNN algorithm shows the attack network life time of 50% higher than MOA-PSO algorithm and 95% higher than GA-SVM algorithm. At attack 2, the proposed CCO-RNN algorithm shows the attack network life time of 70% higher than MOA-PSO algorithm and 96% higher than GA-SVM algorithm. At attack 3, the proposed CCO-RNN algorithm shows the attack network life time of 50% higher than MOA-PSO algorithm and 95% higher than GA-SVM algorithm. At attack 4, the proposed CCO-RNN algorithm shows the attack network life time of 80% higher than MOA-PSO algorithm and 98% higher than GA-SVM algorithm. At attack 5, the proposed CCO-RNN algorithm shows the attack network life time of 75% higher than MOA-PSO algorithm and 97% higher than GA-SVM algorithm.

6. CONCLUSION

In this manuscript, a Chaotic Coyote Optimize Recurrent Neural Networks (CSA-CCO-RNN) classifier is proposed for detecting the Cyber Security attacks in cloud computing. Sincecyber Security attack detection is a very challenging problem in cloud computing.In this method, to classify the process performances we have used text representation method by using system call trace used by classical classifiers. A categorization of system call information has been utilized to differentiate the behaviour amid the normal and attack types. Here, the proposed RNN model which accomplished well on NSL-KDD dataset. The efficiency of the CSA-CCO-RNN method is compared with two existing methods, CSA-MOA-PSO and CSA-GA-SVM. The proposed CSA-CCO-RNN shows the node delivery ratio of 6.35% higher than CSA-MOA-PSO and 3.69% higher than CSA-GA-SVM and the proposed CSA-CCO-RNN shows the attack delivery ratio of 4.12% higher than CSA-MOA-PSO and 7.34% higher than CSA-GA-SVM. The proposed CSA-CCO-RNN shows the attack energy consumption of 3.27% lower than CSA-MOA-PSO and 3.52% lower than CSA-GA-SVM. The proposed CSA-CCO-RNN shows the attack network life time of 81% higher than CSA-MOA-PSO and 97% higher than CSA-GA-SVM method. The experimental outcomes demonstrate the CSA-CCO-RNN method is more efficient likened to other existing methods.

REFERENCES

[1] A. Ghosal and S. Halder, **A survey on energy efficient intrusion detection in wireless sensor networks**, Journal of Ambient Intelligence and Smart Environments, vol. 9, no. 2, pp. 239–261, 2017.

[2] B. B. Zarpelão, R. S. Miani, C. T. Kawakani, and S. C. de Alvarenga, **A survey of intrusion detection in Internet of Things**, Journal of Network and Computer Applications, vol. 84, pp. 25–37, 2017. References

[3] F. Zhang, H. Kodituwakku, J. Hines and J. Coble, **Multilayer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on**

Network, System, and Process Data, *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4362-4369, 2019. Available: 10.1109/tii.2019.2891261.

[4] C. Fluke and C. Jacobs, **Surveying the reach and maturity of machine learning and artificial intelligence in astronomy**, *WIREs Data Mining and Knowledge Discovery*, vol. 10, no. 2, 2019. Available: 10.1002/widm.1349 [Accessed 22 February 2021].

[5] P. Sharma, K. Choudhary, K. Gupta, R. Chawla, D. Gupta and A. Sharma**, Artificial plant optimization algorithm to detect heart rate & presence of heart disease using machine learning,** *Artificial Intelligence in Medicine*, vol. 102, p. 101752, 2020. Available: 10.1016/j.artmed.2019.101752 [Accessed 22 February 2021].

[6] D. Dimiduk, E. Holm and S. Niezgoda, **Perspectives on the Impact of Machine Learning, Deep Learning, and Artificial Intelligence on Materials, Processes, and Structures Engineering,** *Integrating Materials and Manufacturing Innovation*, vol. 7, no. 3, pp. 157-172, 2018. Available: 10.1007/s40192-018-0117-8 [Accessed 22 February 2021].

[7] A. Abduvaliyev, A. Pathan, Jianying Zhou, R. Roman and Wai-Choong Wong, **On the Vital Areas of Intrusion Detection Systems in Wireless Sensor Networks,** *IEEE Communications Surveys & Tutorials*, vol. 15, no. 3, pp. 1223-1237, 2013. Available: 10.1109/surv.2012.121912.00006.

[8] P. Mishra, V. Varadharajan, U. Tupakula and E. Pilli, **A Detailed Investigation and Analysis of Using Machine Learning Techniques for Intrusion Detection**, *IEEE Communications Surveys & Tutorials*, vol. 21, no. 1, pp. 686-728, 2019. Available: 10.1109/comst.2018.2847722.

[9] M. Mazini, B. Shirazi and I. Mahdavi, **Anomaly network-based intrusion detection system using a reliable hybrid artificial bee colony and AdaBoost algorithms,** *Journal of King Saud University - Computer and Information Sciences*, vol. 31, no. 4, pp. 541-553, 2019. Available: 10.1016/j.jksuci.2018.03.011

[10] P. Kaur, M. Kumar and A. Bhandari, **A review of detection approaches for distributed denial of service attacks,** *Systems Science & Control Engineering*, vol. 5, no. 1, pp. 301-320, 2017. Available: 10.1080/21642583.2017.1331768

[11] R. Vinayakumar, M. Alazab, K. Soman, P. Poornachandran, A. Al-Nemrat and S. Venkatraman, **Deep Learning Approach for Intelligent Intrusion Detection System**, *IEEE Access*, vol. 7, pp. 41525-41550, 2019. Available: 10.1109/access.2019.2895334

[12] V. Alaparthy and S. Morgera, **A Multi-Level Intrusion Detection System for Wireless Sensor Networks Based on Immune Theory**, *IEEE Access*, vol. 6, pp. 47364-47373, 2018. Available: 10.1109/access.2018.2866962.

[13] D. Ding, Q. Han, Y. Xiang, X. Ge and X. Zhang, **A survey on security control and attack detection for industrial cyber-physical systems**, *Neurocomputing*, vol. 275, pp. 1674-1683, 2018. Available: 10.1016/j.neucom.2017.10.009.

[14] S. Sahoo, S. Mishra, J. Peng and T. Dragicevic, **A Stealth Cyber-Attack Detection Strategy for DC Microgrids,** *IEEE Transactions on Power Electronics*, vol. 34, no. 8, pp. 8162-8174, 2019. Available: 10.1109/tpel.2018.2879886.

[15] F. Zhang, H. Kodituwakku, J. Hines and J. Coble, **Multilayer Data-Driven Cyber-Attack Detection System for Industrial Control Systems Based on Network, System, and Process Data,** *IEEE Transactions on Industrial Informatics*, vol. 15, no. 7, pp. 4362-4369, 2019. Available: 10.1109/tii.2019.2891261.

[16] I. Poornima and B. Paramasivan, **Anomaly detection in wireless sensor network using machine learning algorithm**, *Computer Communications*, vol. 151, pp. 331-337, 2020. Available: 10.1016/j.comcom.2020.01.005.

[17] W. Wang et al., **HAST-IDS: Learning Hierarchical Spatial-Temporal Features Using Deep Neural Networks to Improve Intrusion Detection,** *IEEE Access*, vol. 6, pp. 1792-1806, 2018. Available: 10.1109/access.2017.2780250. References

[18] Y. Soe, P. Santosa and R. Hartanto, **DDoS Attack Detection Based on Simple ANN with SMOTE for IoT Environment**, *2019 Fourth International Conference on Informatics and Computing (ICIC)*, 2019. Available: 10.1109/icic47613.2019.8985853.

[19] A. Abusitta, M. Bellaiche and M. Dagenais, **An SVM-based framework for detecting DoS attacks in virtualized clouds under changing environment,** *Journal of Cloud Computing*, vol. 7, no. 1, 2018. Available: 10.1186/s13677-018-0109-4

[20] M. Zekri, S. Kafhali, N. Aboutabit and Y. Saadi**, DDoS attack detection using machine learning techniques in cloud computing environments**, *2017 3rd International Conference of Cloud Computing Technologies and Applications (CloudTech)*, 2017. Available: 10.1109/cloudtech.2017.8284731.

[21] J. Pierezan, L. dos Santos Coelho, V. Cocco Mariani, E. Hochsteiner de Vasconcelos Segundo and D. Prayogo, **Chaotic coyote algorithm applied to truss optimization problems,** *Computers & Structures*, vol. 242, p. 106353, 2021. Available: 10.1016/j.compstruc.2020.106353.

[22] Z. Pang, F. Niu and Z. O'Neill, **Solar radiation prediction using recurrent neural network and artificial neural network: A case study with comparisons**, *Renewable Energy*, vol. 156, pp. 279-289, 2020. Available: 10.1016/j.renene.2020.04.042.

[23] S. Sandosh, V. Govindasamy and G. Akila, **Enhanced intrusion detection system via agent clustering and**

classification based on outlier detection, *Peer-to-Peer Networking and Applications*, vol. 13, no. 3, pp. 1038-1045, 2020. Available: 10.1007/s12083-019-00822-3.

[24] P. Tao, Z. Sun and Z. Sun, **An Improved Intrusion Detection Algorithm Based on GA and SVM,** *IEEE Access*, vol. 6, pp. 13624-13631, 2018. Available: 10.1109/access.2018.2810198.

[25] M. Ferrag, L. Maglaras, S. Moschoyiannis and H. Janicke, **Deep learning for cyber security intrusion detection: Approaches, datasets, and comparative study**, *Journal of Information Security and Applications*, vol. 50, p. 102419, 2020. Available: 10.1016/j.jisa.2019.102419.