Proposed algorithm base optimisation plan for feature selection-based intrusion detection in cloud computing

Imane Laassar[1], Moulay Youssef Hadi[1], Arifullah[2], Hassnae Remmach[3], Fawad Salam Khan[2]
[1]Department of Computer Science, Faculty of Computer Sciences and Informatics, Université Ibn Tofail Morocco, Kenitra, Morocco
[2]Department of Computer Science, Faculty of Computing and Artificial Intelligent Air University, Islamabad, Pakistan
[3]LAMIGEP, EMSI Marrakesh, Morocco

Article Info

Article history:

Received Jul 15, 2023
Revised Nov 4, 2023
Accepted Dec 13, 2023

Keywords:

Classifiers
Cloud computing
Communication
Convergence
Intrusion detection system

ABSTRACT

A crucial element in detecting unusual network system behavior is the network intrusion detection system (NIDS), which also helps to stop network attacks from happening. Despite the fact that a great deal of machine learning techniques has been used in intrusion detection, current solutions still struggle to provide accurate classification results. Furthermore, when dealing with imbalanced multi-category traffic data, a single classifier may not be able to produce a superior. Particularly, internet of things (IoT) gadgets is now a commonplace aspect of life. On the other hand, some problems are becoming worse and lack clear remedies. Convergence, communication speed, and security between various IoT devices are among the primary concerns. In order to achieve this goal, an enhanced artificial bee colony technique utilizing binary search equations and neural networks—known as the (BABCN) algorithm for intrusion detection in terms of convergence and communication speed—is presented in this study. The artificial bee is improved by the depth-first search framework and binary search equations upon which the BABCN method is based. The suggested approach has a good ability to detect intrusions in the network and enhances categorization, according to the findings obtained by using the NSL-KDD dataset.

Corresponding Author:

Imane Laassar
Department of Computer Science, Faculty of Computer Sciences and Informatics
Université Ibn Tofail Morocco
Kenitra, Morocco
Email: imane.laassar@uit.ac.ma

1. INTRODUCTION

The internet is now a part of almost every area of modern life, thus the number of devices connected to it is growing quickly. In particular, internet of things (IoT) devices are becoming more and more ubiquitous in daily life. But some problems are getting worse, and several researchers are also talking about how to fix them [1], [2]. Intrusion detection is a technology used in cloud and IoT security methods to locate, confirm, and stop unauthorised access to a computer network or internetwork. There are significant network confidentiality conflicts to be resolved because of the remarkable advancements in data technology. As a result, an intrusion detection system (IDS) is essential for ensuring network security [3]. IDS are categorised into multiple unique techniques. There are two main categories: active and inactive. Threats that are newly appearing cannot be handled by the conventional active IDS. Finding and differentiating between regular and anomalous network connections is one of the main obstacles in detecting intrusions because of the vast number of components and properties of this type of network. IDS is widely used to identify the location and mode of intrusions. To achieve real-time intrusion detection, the scientists thoroughly investigated a number of element selection methodologies [4], [5]. Reducing the amount of features by choosing only the most important ones is a strong case for increasing the speed and accuracy of categorization algorithms. Machine learning algorithms are frequently used to identify different kinds of attacks, and they can help network administrators respond to attacks by pointing them in the direction of the most effective course of action. However, most of these traditional machine learning methods fall into the shallow learning category and necessitate a thorough feature extraction and selection process [6], [7]. Finding and differentiating between regular and anomalous network connections is one of the main obstacles in detecting intrusions because of the vast number of components and properties of this type of network. IDS is widely used to identify the location and mode of intrusions. The core element of an IDS is the classifier, which employs a detection algorithm to differentiate between incursion and regular activities. Implementing a classifier with an accurate detection mechanism can be challenging, particularly in networks of cloud computing and the Internet of Things that have a large number of devices [8], [9]. The functioning conditions and structure of IoT and CC integration are shown in Figure 1.

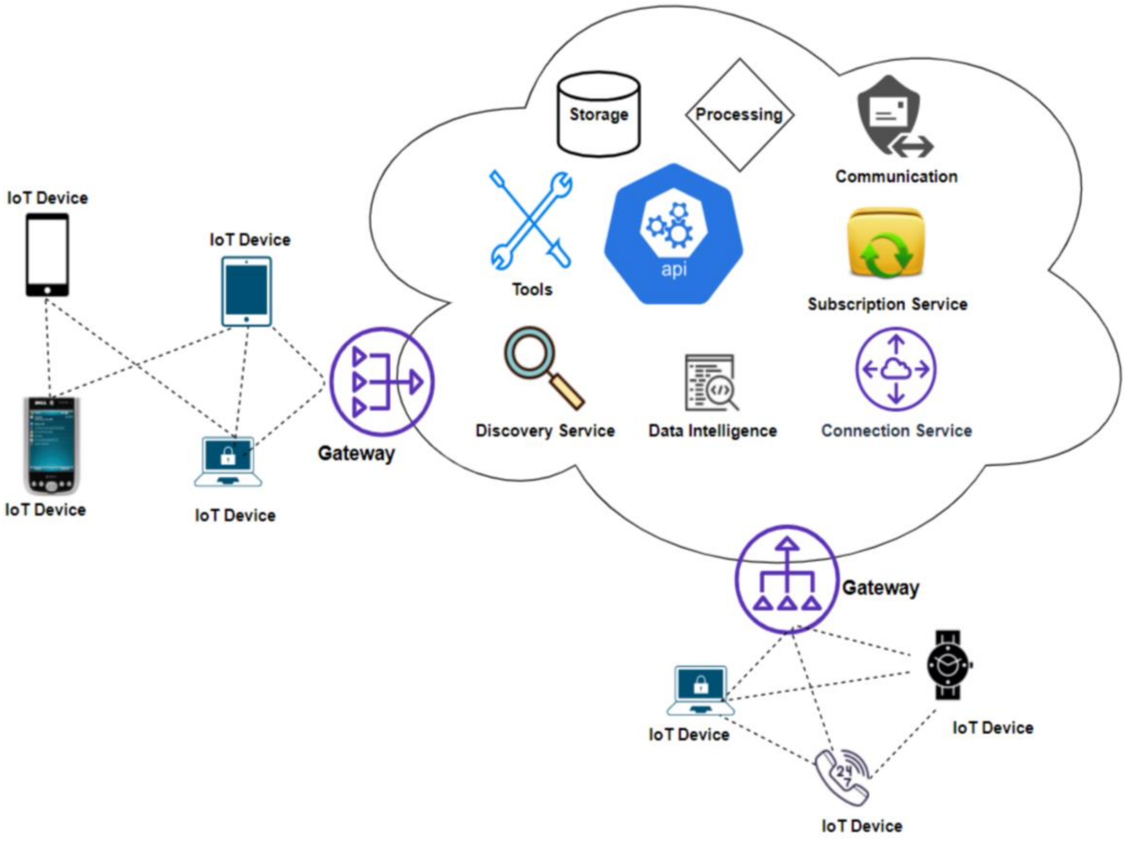

Figure 1. Structure of CC and IoT [10]

The remainder of this essay is organised as follows: information regarding related work is presented in section 2. The suggested method is covered in section 3, the parameters are covered in section 4. The results are shown in section 5, and the conclusion is discussed in Section 6.

2. RELATED WORK

Anomaly-based and signature-based intrusion detection approaches are the two primary categories. When using signature-based techniques, the system stores a number of preconfigured signatures that have been evaluated and shown to be effective against intrusion patterns. The algorithm also matches the activities performed with these patterns; if a pattern is detected, the action will be classified as an incursion. Of course, these methods are unable to detect zero-day or newly discovered threats. On the other hand, these methods excel at recognising identified hazards and their trends [11], [12]. Using anomaly-based techniques, a vision of typical activity is created, and an anomaly may then indicate an incursion. It is well knowledge that abnormal incursions are very difficult to identify since there is no predetermined routine for monitoring. An event is usually considered abnormal if it happens much more or less often than a predetermined threshold [13], [14]. Certain AI techniques use tree-based algorithms, such as random forests and decision trees, to create a framework for effectively identifying infiltration. A decision tree algorithm makes decisions in accordance with the problem's parameters one step at a time. To model an issue, a decision tree might not be enough in every situation. Consequently, random forest algorithms use many decision trees in order to increase the overall accuracy of decision-making.

Awan *et al.* [15], B. Akay and D. Karaboga [16] has presented an anomaly-based method (IDSML) for software-defined networks that combines multiple different tree-based algorithms to improve detection performance. In other AI techniques, neural networks are used to precisely identify if a certain event resembles recognised patterns. Networks of neurons are composed of several interconnected nodes with pattern recognition skills. Due to the numerous parameters involved in decision-making problems, computations in a neural network take a long time in [17], [18]. In several research, the main technique for detection has been neural networks. Karaboga developed the artificial bee colony algorithm in 2005 as a heuristic swarm intelligence system designed to mimic honeybee group behaviour. It was first developed to solve some numerical optimisation problems. The artificial bee colony algorithm (ABCA) was applied to multivariate function optimisation in [19] and contrasted with other techniques such as particle swarm optimisation (PSO) and genetic algorithm (GA). The outcomes demonstrate that ABC is the best algorithm out there. As opposed to that, the artificial bee colony algorithm is good at exploring the solution but suffers from exploitation and tends to settle into a local optimum. An improvement on the ABC method was presented with the GABC algorithm [20], which improves exploitation by incorporating information on the global optimal solution into the solution search equation. A method known as the multi-strategy ensemble artificial bee colony (MEABC) was proposed in [21]. Throughout the search process, a range of distinct solution search strategies coexist and vie for progeny in MEABC. When the MEABC technique is used for continuous optimisation problems, ABC performs noticeably better. DFSABC Elite, an artificial bee colony method with a depth-first design and Elite-Guided Search Equations, was introduced in [22], [23]. The exploitability of the algorithm is enhanced by allocating more computational resources to the best possible solutions [24].

3. PROPOSED ALGORITHM

The iterative, population-based ABC technique is a potent strategy for solving problems involving numerical optimisation. The previously referenced papers are [25]. For exploration, equations are more powerful than for exploitation. Furthermore, the convergence performance of the ABC method is not very good. To better balance exploration and exploitation, a Binary Search (BSF) architecture and two search equation solutions, as shown in (1), were therefore proposed in [26], [27]. The BSF procedure is employed to increase the algorithm's exploitability. When assigning more computing resources, the BSF framework can prioritise better solutions more highly. Training of the algorithm is accelerated by the search equations that keep the solution with the highest fitness value on each iteration [28], [29].

$$V_{i,j} = X_{e,j} + \phi_{e,j} \times \left(X_{e,j} - X_{k,j}\right) \tag{1}$$

$$V_{e,j} = \frac{1}{2}\left(X_{e,j} + X_{best,j}\right) + \phi_{e,j} \times \left(X_{best,j} - X_{k,j}\right) \tag{2}$$

Where the binary search solution and the current population were used to randomly select the solutions X_e and X_k, respectively. There is no equivalence between e and k. Right now, X_best is the best option. In the interval [-1, 1], there exist two random real numbers, i, j and e, j. The issue that the candidate solution search equation in paper [30], [31] has an excessively significant disruption to the search solution is addressed in paper [30] in an effort to better balance ABC's exploration and exploitation capacities. After that, a binary search equation is shown. For the accepted solutions and the candidate solutions, separate search equations should be applied. In the present population, X_k represents a randomly selected solution, and X_e represents a randomly selected solution from the solution for binary search. Terms e and k cannot be used interchangeably. As of right moment, the best choice is X_best. I, j and e, j are two random real variables in the interval [-1, 1]. In order to better balance ABC's exploration and exploitation capacities, the problem that the candidate solution search equation in paper [32] has an overly severe disruption to the search solution is addressed in article [20]. Next, a binary search equation is shown. For the accepted answers and the candidate solutions, separate search equations should be used [33], [34].

$$P_i = \frac{c_1 \times pbest_i + c_2 \times gbest}{c_1 + c_2} \tag{3}$$

$$X_i = N\left(\frac{gbest+pbest_i}{2}, \square gbest - pbest_i \square\right) \tag{4}$$

The Gaussian distribution in this case is denoted by N, the mean by gbest+ [pbest]_i, and the standard deviation by gbest, [pbest]_i. We utilise the Gaussian distribution in (3) to benefit from the data surrounding pbest and gbest. A similar Gaussian search equation is proposed [35], [36], based on (4).

$$V_{e,j} = N\left(\frac{X_{best,j}+X_{i,j}}{2}, \square X_{best,j} - X_{i,j} \square\right) \tag{5}$$

$$V_{e,j} = \frac{1}{2}\left(X_{e,j} + X_{best,j}\right) + \phi(X_{e,j} + X_{best,j}) + \phi_{e,j}(X_{best,j} - X_{e,j}) \tag{6}$$

$$net_j = \sum_{i=1}^{m} \omega_{i,j}\chi_i + \theta_i \tag{7}$$

Finally, the neural network is trained using the back propagation method using the initial weight and threshold values produced by the BABCN algorithm. The back propagation method uses gradient descent in an attempt to reduce the training error. The neural network's parameters for network traffic intrusion detection will be determined by utilising the weights and thresholds that have the lowest training error [37], [38]. The following are the working criteria for the neural network and suggested back propagation: After selecting a training sample of data, the weight values for the connections (ω_jk) between the hidden layer neurons and the output layer neurons and (ω_j) between the hidden layer neurons and the input layer neurons are generated at random. Create the j threshold values for the neurons in the hidden layer as well [39], [40].

$$V_{e,j} = \frac{1}{2}\left(X_{e,j} + X_{best,j}\right) + \phi(X_{e,j} + X_{best,j}) + \phi_{e,j}(X_{best,j} - X_{e,j}) \tag{8}$$

$$net_j = \sum_{i=1}^{m} \omega_{i,j}\chi_i + \theta_i \tag{9}$$

$$y_j = \vartheta_1\left(net_j\right) \tag{10}$$

$$net_k = \sum_{j=1}^{h} \omega_{jk}y_j + \theta_k \tag{11}$$

$$Z_k = \vartheta_2(net_k) \tag{12}$$

The error of the neural network is evaluated using (8). If the mistake meets the requirements, in (9) and (10) are applied; if not, (11) and (12) are applied [41], [42].

$$J(w) = \frac{1}{2}\sum_{k=1}^{q}(t_k - z_k)^2 \tag{13}$$

In (13) and (14), when applied to the output layer and hidden layer, modify the threshold and weight values. The weight and threshold settings between the input layer and the hidden layer are changed using (15) [43], [44].

$$\nabla\omega_{jk} = \eta(t_k - z_k)\vartheta_2{}'(net_k)y_j \tag{14}$$

$$\nabla\theta_k = \eta(t_k - z_k)\vartheta_2{}'(net_k) \tag{15}$$

$$\nabla\omega_{i,j} = \eta\left[\sum_{k=1}^{q} \omega_{jk}\delta_k\right]\vartheta_1{}'\left(net_j\right)\chi_i \tag{16}$$

$$\nabla\theta_j = \eta\left[\sum_{k=1}^{q} \omega_{jk}\delta_k\right]\vartheta_1{}'\left(net_j\right) \tag{17}$$

$$\delta_k = -\frac{\partial J}{\partial net_k} = -\frac{\partial J}{\partial z_k}\frac{\partial z_k}{\partial net_k} = (t_k - z_k)\vartheta_2{}'(net_k) \tag{18}$$

The outcomes of (16) [45], [46] can be used to derive the new weight standards ω_ij and the new threshold values j between the input layer and the hidden layer, as well as the new weight values ω_jk and the new threshold values k between the hidden layer and the output layer. The new weight values ω_ij and the new threshold values j between the input layer and the hidden layer, as well as the new weight values ω_jk and the new threshold values k between the hidden layer and the output layer, may be recovered after learning the findings from (17) [47], [48].

$$fit_i = \begin{cases} \frac{1}{1+f(X_i)} f(X_i) \geq 0 \\ 1+\square f(X_i)\square f(X_i) < 0 \end{cases} \quad (19)$$

With the adjusted weight and threshold values, rerun step and (18). If the fault matches the specifications, end the training process. If not, use the current weights and thresholds as neural work input signals to retrieve the pertinent output signal from the neural network. The loss function of a neural network (19), is the target function of (18). Determine the appropriate max cycle number (MCN) [49], [50].

4. EVALUATION METRICS

The following evaluation parameters are measured in this paper, which are as (20).

$$AC = \frac{TP+TN}{TP+TN+FP+FN} \quad (20)$$

Accuracy (AC) is defined by (20) as the proportion of samples that have been correctly identified to all samples (41) [45].

$$TPR = \frac{TP}{TP+FN} \quad (21)$$

The true positive rate (TPR), which is the percentage of correctly identified anomaly samples over all anomaly samples, is equal to the detection rate (DR) [51], [52].

$$FPR = \frac{FP}{FP+TN} \quad (22)$$

The ratio of the total number of normal samples to the number of normal samples that were incorrectly labeled as anomaly samples is known as the false positive rate (FPR) [53].

5. RESULTS AND DISCUSSION

The multiclass classification results are compared with the suggested algorithm in Table 1 and Figure 1 present the result of different paramaters are used and these parameters are used in testing process for different purpose section. Comparing the correctness of the model constructed with all the features and the 13 features that were chosen is the main goal of the modelling stage as it is presented in the Intrusion Detection System. The accuracy comparison between all features and 13 features using a Decision Tree classifier is displayed in the accompanying Figure 1.

Table 1. Shows the results of different parameters in classification

Algorithm	AUC DDOS	AUC DOS	AUC reconnaissance	Auc normal	AUC theft
Mlp algorithm	0.98	0.98	0.99	1	0.96
Abc algorithm	0.98	0.98	0.99	1	0.92
Bat algorithm	0.96	0.95	0.98	1	0.93
Proposed BABCN algorithm	0.98	0.98	0.98	1	0.97

Selecting a model's threshold gives rise to categorization problems. Table 1 lists the two ROC curve parameters: the rate of false positives and the true positives. The strongest predictor of a model when it comes to selecting which data to use for classification analysis is its area under the curve (AUC). One example of its application is the ROC curve. In this instance, the true positive rate and false positive rate are contrasted. Table 2 and Figures 2-3 shows the overall good performance of the RF for multiclass classification.

A receiver operating characteristic (ROC) curve shows the performance measurement instrument. Selecting a model's threshold gives rise to categorization problems. This ROC curve has two parameters: the rate of false positives and the true positives. The results for a 32-batch operation are shown in Table 2. In this instance, as the number of study epochs rose, the mean accuracy of the proposed BABCN algorithm classifier decreased. The accuracy dropped as the number of epochs rose from 10 to 32. The batch operation of several algorithms is shown in Figure 4 and Table 3.

Table 2. Metrics batch size 32

Algorithm	Epoch	Mean accuracy	Elapsed time
MLP algorithm	10	0.98	0.99
ABC algorithm	30	0.97	0.98
Bat algorithm	27	0.83	0.94
Proposed BABCN algorithm	32	0.99	0.97

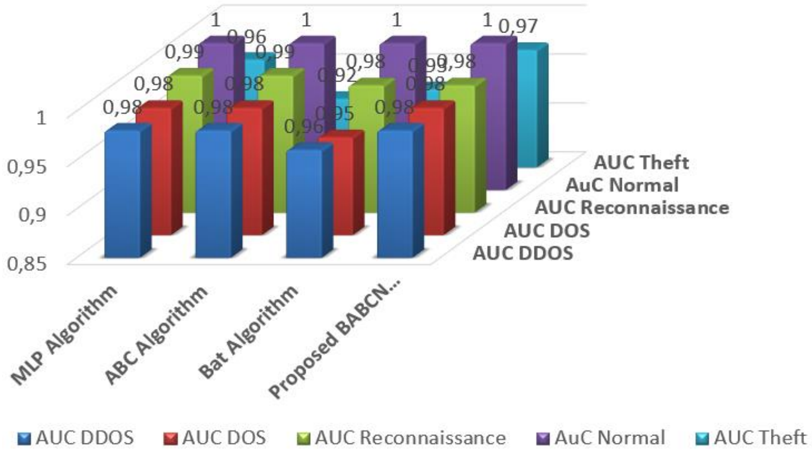

Figure 2. Different parameters in classification result

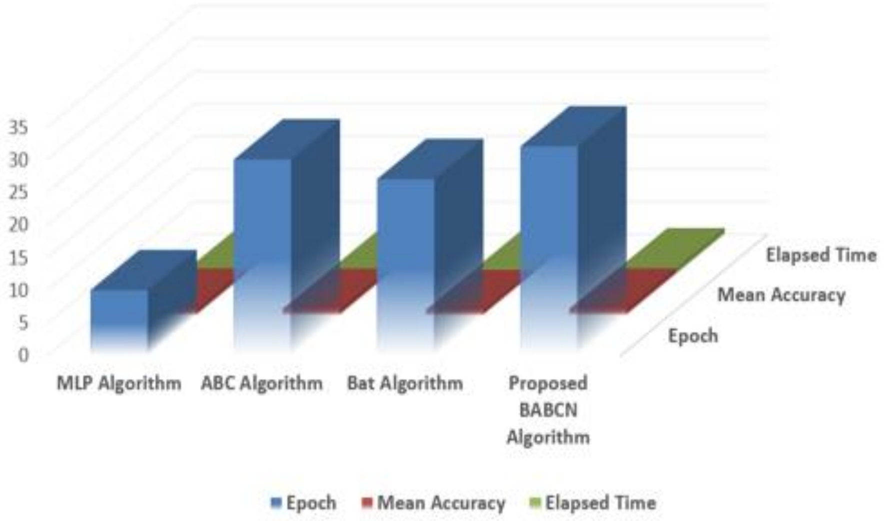

Figure 3. Metrics batch size 32

The results are shown in Tables 3 and 4 for batch sizes of 64 and 128. With an increase in study epochs, the proposed BABCN method classifier's mean accuracy appeared to rise. The BABCN somewhat decreased as the number of epochs climbed from 15 to 45, and then it increased at 45 epochs. Table 4 illustrates that a lower duration time may be achieved with a bigger batch size. Figure 5 and Table 4 shows the accuracy of several methods.

Table 3. Metrics batch size 128

Algorithm	Epoch	Mean accuracy	Elapsed time
MLP algorithm	25	0.98	0.99
ABC algorithm	35	0.97	0.99
Bat algorithm	36	0.89	0.98
Proposed BABCN algorithm	45	0.98.8	0.99

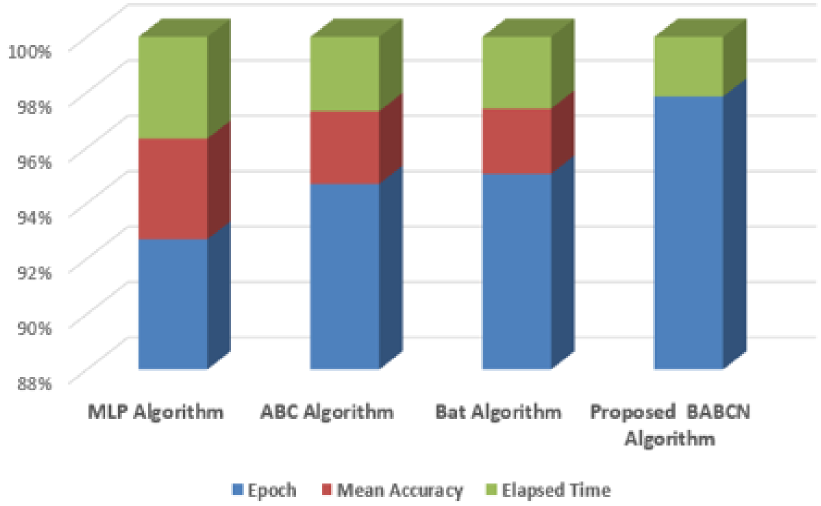

Figure 4. Metrics batch size 128

Table 4. Accuracy of different approaches

Algorithm	Epoch	Mean Accuracy	Elapsed Time
MLP algorithm	35	0.98	0.99
Bat algorithm	37	0.96	0.98
ABC algorithm	40	0.98	0.99
Proposed BABCN algorithm	49	0.99	0.99

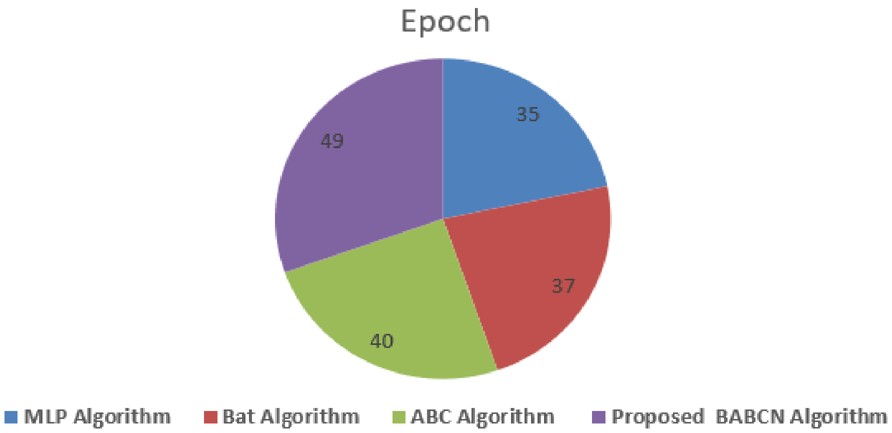

Figure 5. Accuracy of different approaches

6. CONCLUSION

In order to offer a thorough comparison and create more effective IDS, numerous other machine learning methods and feature selection strategies may be used in future research. Since the current IDS can only forecast known assaults, it can also be used to identify novel attacks in other scientific and technological domains. In this study, we examined several deep learning and machine learning methods on an Internet of Things network and compared them with our suggested methodology. We considered RF, CNN, MLP, and the suggested BABCN algorithm analyses. CNN and random forests produced the greatest results in terms of multiclass classification accuracy and AUC. The accuracy increased marginally in trials with 128 batches but declined slightly in trials with 32 and 64 batches as more epochs were added. We also found that increasing the batch size accelerated the computation. Increasing the batch size by two could result in 1.3-2.4 times faster calculation for the proposed BABCN algorithm and 1.8-2.4 times faster computation for CNN. Our extended.

REFERENCES

[1] P. Ehin, M. Solvak, J. Willemson, and P. Vinkel, “Internet voting in Estonia 2005–2019: Evidence from eleven elections,” *Government Information Quarterly*, vol. 39, no. 4, p. 101718, Oct. 2022, doi: 10.1016/j.giq.2022.101718.

[2] N. Chaabouni, M. Mosbah, A. Zemmari, C. Sauvignac, and P. Faruki, “Network intrusion detection for IoT security based on learning techniques,” *IEEE Communications Surveys & Tutorials*, vol. 21, no. 3, pp. 2671–2701, 2019, doi: 10.1109/COMST.2019.2896380.

[3] M. H. Bhuyan, D. K. Bhattacharyya, and J. K. Kalita, “Network anomaly detection: methods, systems and tools,” *IEEE Communications Surveys & Tutorials*, vol. 16, no. 1, pp. 303–336, 2014, doi: 10.1109/SURV.2013.052213.00046.

[4] B. Molina-Coronado, U. Mori, A. Mendiburu, and J. Miguel-Alonso, “Survey of network intrusion detection methods from the perspective of the knowledge discovery in databases process,” *IEEE Transactions on Network and Service Management*, vol. 17, no. 4, pp. 2451–2479, Dec. 2020, doi: 10.1109/TNSM.2020.3016246.

[5] W. Li, “Anti-forensic digital investigation for unauthorized intrusion on a wireless network,” Auckland University of Technology, 2013.

[6] D. Kwon, H. Kim, J. Kim, S. C. Suh, I. Kim, and K. J. Kim, “A survey of deep learning-based network anomaly detection,” *Cluster Computing*, vol. 22, no. S1, pp. 949–961, Jan. 2019, doi: 10.1007/s10586-017-1117-8.

[7] D. A. M. S. Revathi, “A detailed analysis on NSL-KDD dataset using various machine learning techniques for intrusion detection,” *International Journal of Engineering Research & Technology (IJERT)*, vol. 2, no. 12, 2013.

[8] R. Vinayakumar, M. Alazab, K. P. 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, doi: 10.1109/ACCESS.2019.2895334.

[9] M. Lotfollahi, M. J. Siavoshani, R. S. H. Zade, and M. Saberian, “Deep packet: a novel approach for encrypted traffic classification using deep learning,” *Soft Computing*, vol. 24, no. 3, pp. 1999–2012, Feb. 2020, doi: 10.1007/s00500-019-04030-2.

[10] P. Mishra, V. Varadharajan, U. Tupakula, and E. S. 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, doi: 10.1109/COMST.2018.2847722.

[11] D. Karaboga and C. Ozturk, “A novel clustering approach: artificial bee colony (ABC) algorithm,” *Applied Soft Computing*, vol. 11, no. 1, pp. 652–657, Jan. 2011, doi: 10.1016/j.asoc.2009.12.025.

[12] N. Imanian, M. E. Shiri, and P. Moradi, “Velocity based artificial bee colony algorithm for high dimensional continuous optimization problems,” *Engineering Applications of Artificial Intelligence*, vol. 36, pp. 148–163, Nov. 2014, doi: 10.1016/j.engappai.2014.07.012.

[13] W. Zou, Y. Zhu, H. Chen, and X. Sui, “A clustering approach using cooperative artificial bee colony algorithm,” *Discrete Dynamics in Nature and Society*, vol. 2010, pp. 1–16, 2010, doi: 10.1155/2010/459796.

[14] W. Gao and S. Liu, “Improved artificial bee colony algorithm for global optimization,” *Information Processing Letters*, vol. 111, no. 17, pp. 871–882, Sep. 2011, doi: 10.1016/j.ipl.2011.06.002.

[15] S. M. Awan, M. Aslam, Z. A. Khan, and H. Saeed, “An efficient model based on artificial bee colony optimization algorithm with Neural Networks for electric load forecasting,” *Neural Computing and Applications*, vol. 25, no. 7–8, pp. 1967–1978, Dec. 2014, doi: 10.1007/s00521-014-1685-y.

[16] B. Akay and D. Karaboga, “Artificial bee colony algorithm for large-scale problems and engineering design optimization,” *Journal of Intelligent Manufacturing*, vol. 23, no. 4, pp. 1001–1014, Aug. 2012, doi: 10.1007/s10845-010-0393-4.

[17] W.-F. Gao, L.-L. Huang, S.-Y. Liu, and C. Dai, “Artificial bee colony algorithm based on information learning,” *IEEE Transactions on Cybernetics*, vol. 45, no. 12, pp. 2827–2839, Dec. 2015, doi: 10.1109/TCYB.2014.2387067.

[18] D.-H. Tran, M.-Y. Cheng, and M.-T. Cao, “Hybrid multiple objective artificial bee colony with differential evolution for the time–cost–quality tradeoff problem,” *Knowledge-Based Systems*, vol. 74, pp. 176–186, Jan. 2015, doi: 10.1016/j.knosys.2014.11.018.

[19] K. Ermis, A. Erek, and I. Dincer, “Heat transfer analysis of phase change process in a finned-tube thermal energy storage system using artificial neural network,” *International Journal of Heat and Mass Transfer*, vol. 50, no. 15–16, pp. 3163–3175, Jul. 2007, doi: 10.1016/j.ijheatmasstransfer.2006.12.017.

[20] D. J. Hemanth, J. Anitha, A. Naaji, O. Geman, D. E. Popescu, and L. H. Son, “A modified deep convolutional neural network for abnormal brain image classification,” *IEEE Access*, vol. 7, pp. 4275–4283, 2019, doi: 10.1109/ACCESS.2018.2885639.

[21] H. Il Park and S. R. Lee, “Evaluation of the compression index of soils using an artificial neural network,” *Computers and Geotechnics*, vol. 38, no. 4, pp. 472–481, Jun. 2011, doi: 10.1016/j.compgeo.2011.02.011.

[22] C. Sun, M. Ma, Z. Zhao, S. Tian, R. Yan, and X. Chen, “Deep transfer learning based on sparse autoencoder for remaining useful life prediction of tool in manufacturing,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2416–2425, Apr. 2019, doi: 10.1109/TII.2018.2881543.

[23] M. Khandelwal and T. N. Singh, “Prediction of blast induced ground vibrations and frequency in opencast mine: A neural network approach,” *Journal of Sound and Vibration*, vol. 289, no. 4–5, pp. 711–725, Feb. 2006, doi: 10.1016/j.jsv.2005.02.044.

[24] H. Fang, M. Rais-Rohani, Z. Liu, and M. F. Horstemeyer, “A comparative study of metamodeling methods for multiobjective crashworthiness optimization,” *Computers & Structures*, vol. 83, no. 25–26, pp. 2121–2136, Sep. 2005, doi: 10.1016/j.compstruc.2005.02.025.

[25] M. Lavertu *et al.*, “A validated 1H NMR method for the determination of the degree of deacetylation of chitosan,” *Journal of Pharmaceutical and Biomedical Analysis*, vol. 32, no. 6, pp. 1149–1158, Aug. 2003, doi: 10.1016/S0731-7085(03)00155-9.

[26] G. Pacini and R. N. Bergman, “MINMOD: a computer program to calculate insulin sensitivity and pancreatic responsivity from the frequently sampled intravenous glucose tolerance test,” *Computer Methods and Programs in Biomedicine*, vol. 23, no. 2, pp. 113–122, Oct. 1986, doi: 10.1016/0169-2607(86)90106-9.

[27] A. W. Sandvik, “Finite-size scaling of the ground-state parameters of the two-dimensional Heisenberg model,” *Physical Review B*, vol. 56, no. 18, pp. 11678–11690, Nov. 1997, doi: 10.1103/PhysRevB.56.11678.

[28] J. Veitch *et al.*, “Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library,” *Physical Review D*, vol. 91, no. 4, p. 042003, Feb. 2015, doi: 10.1103/PhysRevD.91.042003.

[29] K. Motz, H. Sterba, and A. Pommerening, “Sampling measures of tree diversity,” *Forest Ecology and Management*, vol. 260, no. 11, pp. 1985–1996, Nov. 2010, doi: 10.1016/j.foreco.2010.08.046.

[30] Q. Li, Q. Meng, J. Cai, H. Yoshino, and A. Mochida, “Predicting hourly cooling load in the building: A comparison of support vector machine and different artificial neural networks,” *Energy Conversion and Management*, vol. 50, no. 1, pp. 90–96, Jan. 2009, doi: 10.1016/j.enconman.2008.08.033.

[31] D. Sebai and A. U. Shah, “Semantic-oriented learning-based image compression by Only-Train-Once quantized autoencoders,” *Signal, Image and Video Processing*, vol. 17, no. 1, pp. 285–293, Feb. 2023, doi: 10.1007/s11760-022-02231-1.

[32] H. Aznaoui, S. Raghay, A. Ullah, and M. H. Khan, “Energy efficient dtrategy for WSN technology using modified HGAF technique,” *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 17, no. 06, p. 4, Jun. 2021, doi: 10.3991/ijoe.v17i06.17739.

[33] H. Aznaoui, A. Ullah, S. Raghay, L. Aziz, and M. H. Khan, “New efficient GAF routing protocol using an optimized weighted sum model in WSN,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 22, no. 1, p. 396, Apr. 2021, doi: 10.11591/ijeecs.v22.i1.pp396-406.

[34] S. Ouhame, Y. Hadi, and A. Arifullah, “A hybrid grey wolf optimizer and artificial bee colony algorithm used for improvement in resource allocation system for cloud technology,” *International Journal of Online and Biomedical Engineering (iJOE)*, vol. 16, no. 14, p. 4, Nov. 2020, doi: 10.3991/ijoe.v16i14.16623.

[35] M. Faheem, U. Akram, I. Khan, S. Naqeeb, A. Shahzad, and A. Ullah, “Cloud computing environment and security challenges: a review,” *International Journal of Advanced Computer Science and Applications*, vol. 8, no. 10, 2017, doi: 10.14569/ijacsa.2017.081025.

[36] S. Umar, S. Baseer, and Arifullah, “Perception of cloud computing in universities of Peshawar, Pakistan,” in *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, IEEE, Aug. 2016, pp. 87–91. doi: 10.1109/INTECH.2016.7845046.

[37] Arifullah, S. Baseer, and S. Umar, “Role of cooperation in energy minimization in visual sensor network,” in *2016 Sixth International Conference on Innovative Computing Technology (INTECH)*, IEEE, Aug. 2016, pp. 447–452. doi: 10.1109/INTECH.2016.7845026.

[38] S. N. Khan *et al.*, “Comparative analysis for heart disease prediction,” *JOIV : International Journal on Informatics Visualization*, vol. 1, no. 4–2, p. 227, Nov. 2017, doi: 10.30630/joiv.1.4-2.66.

[39] A. Ullah, “Artificial bee colony algorithm used for load balancing in cloud computing: review,” *IAES International Journal of Artificial Intelligence (IJ-AI)*, vol. 8, no. 2, p. 156, Jun. 2019, doi: 10.11591/ijai.v8.i2.pp156-167.

[40] A. Ullah, N. M. Nawi, and M. H. Khan, “BAT algorithm used for load balancing purpose in cloud computing: an overview,” *International Journal of High Performance Computing and Networking*, vol. 16, no. 1, p. 43, 2020, doi: 10.1504/IJHPCN.2020.110258.

[41] A. Ullah, N. M. Nawi, and S. Ouhame, “Recent advancement in VM task allocation system for cloud computing: review from 2015 to2021,” *Artificial Intelligence Review*, vol. 55, no. 3, pp. 2529–2573, Mar. 2022, doi: 10.1007/s10462-021-10071-7.

[42] A. Ullah and N. M. Nawi, “An improved in tasks allocation system for virtual machines in cloud computing using HBAC algorithm,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 4, pp. 3713–3726, Apr. 2023, doi: 10.1007/s12652-021-03496-z.

[43] T. Alam, A. Ullah, and M. Benaida, “Deep reinforcement learning approach for computation offloading in blockchain-enabled communications systems,” *Journal of Ambient Intelligence and Humanized Computing*, vol. 14, no. 8, pp. 9959–9972, Aug. 2023, doi: 10.1007/s12652-021-03663-2.

[44] A. Ullah, A. Salam, H. El Raoui, D. Sebai, and M. Rafie, “Towards more accurate iris recognition system by using hybrid approach for feature extraction along with classifier,” *International Journal of Reconfigurable and Embedded Systems (IJRES)*, vol. 11, no. 1, p. 59, Mar. 2022, doi: 10.11591/ijres.v11.i1.pp59-70.

[45] A. Ullah and A. Chakir, “Improvement for tasks allocation system in VM for cloud datacenter using modified bat algorithm,” *Multimedia Tools and Applications*, vol. 81, no. 20, pp. 29443–29457, Aug. 2022, doi: 10.1007/s11042-022-12904-1.

[46] A. Ullah, I. Laassar, C. B. Şahin, O. B. Dinle, and H. Aznaoui, “Cloud and internet-of-things secure integration along with security concerns,” *International Journal of Informatics and Communication Technology (IJ-ICT)*, vol. 12, no. 1, p. 62, Apr. 2023, doi: 10.11591/ijict.v12i1.pp62-71.

[47] M. F. Falah *et al.*, “Comparison of cloud computing providers for development of big data and internet of things application,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 22, no. 3, p. 1723, Jun. 2021, doi: 10.11591/ijeecs.v22.i3.pp1723-1730.

[48] A. A. Aziz, S. Osman, S. Widyarto, S. Marjudi, N. R. M. Suradi, and R. Handan, “Quantifying quantitative correlation of provider selection influences cloud security,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 31, no. 3, p. 1642, Sep. 2023, doi: 10.11591/ijeecs.v31.i3.pp1642-1647.

[49] I. Odun-Ayo, T.-A. Williams, and J. Yahaya, “Cloud management and monitoring - a systematic mapping study,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 21, no. 3, p. 1648, Mar. 2021, doi: 10.11591/ijeecs.v21.i3.pp1648-1662.

[50] N. Bansal and A. K. Singh, “Effective task scheduling algorithm in cloud computing with quality of service alert bees and grey wolf optimization,” *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 25, no. 1, p. 550, Jan. 2022, doi: 10.11591/ijeecs.v25.i1.pp550-560.

[51] S. N. Khan *et al.* “Comparative analysis for heart disease prediction,” *JOIV: International Journal on Informatics Visualization*, vol. 1, no. 4-2, pp. 227-231, 2017.

[52] G. Suciu, S. Halunga, A. Ochian, and V. Suciu, “Network management and monitoring for cloud systems,” in *Proceedings of the 2014 6th International Conference on Electronics, Computers and Artificial Intelligence (ECAI)*, IEEE, Oct. 2014, pp. 1–4. doi: 10.1109/ECAI.2014.7090169.

[53] K. Alhamazani *et al.*, “An overview of the commercial cloud monitoring tools: research dimensions, design issues, and state-of-the-art,” *Computing*, vol. 97, no. 4, pp. 357–377, Apr. 2015, doi: 10.1007/s00607-014-0398-5.

BIOGRAPHIES OF AUTHORS

Imane Laassar working as Research Assistant at Department of Computer Science, Faculty of Science, IbnTofail University, Kenitra, Morocco. Her main research interests focus on Cloud computing, artificial intelligence, metaheuristic modeling and optimization, evolutionary computations, and optimization algorithms. She can be contacted at email: imane.laassar@uit.ac.ma.

Moulay Youssef Hadi working as professor of higher education (Full Professor) at the Ibn Tofail University of Kenitra-Morocco since 2009. He also holds the position of Deputy Director in charge of educational affairs at the Higher School of Technology of Kénitra. His main research interests focus on cloud computing, artificial intelligence, and optimization algorithms. He can be contacted at email: hadi@uit.ac.ma.

Arifullah he Completed his Ph.D. in Cloud computing with 2 years of experience in Teaching and Research. His area of expertise in cloud computing, IoT. Areas of interest include Software Defined Networking (SDN), Load Balancing, switches migration, WSN, E-Learning, AI, WSN, and security. Currenty working as Assistant Professior in Air University Pakistan. He can be contacted at email: arifullah@mail.au.edu.pk.

Hassnae Remmach she received Ph.D. in Computer Science at Cadi Ayyad University and current working as Assistant professor at LAMIGEP, EMSI Marrakesh, Morocco. She can be contacted at email: remmach.hassnae@gmail.com.

Dr. Fawad Salam Khan completed Ph.D. in Electrical Engineering from Universiti Tun Hussain Onn Malaysia. He is currently serving as Assistant Professor in FCAI Air University, Islamabad. He received various gold and silver medals for different AI projects in Malaysia. He is an IEEE graduate student member and professional member of PEC Pakistan, completed BS Computer Engineering from SSUET Pakistan in 2002, ME Computer Engineering from NEDUET Pakistan in 2011, and MS Computer Science from IIUI Pakistan in 2020. He can be contacted at email: he190038@siswa.uthm.edu.my.