Proposed algorithm base optimization scheme for intrusion detection using feature selection

Imane Laassar, Moulay Youssef Hadi
Department of Computer Science, Faculty of Computer Sciences and Informatics, Université Ibn Tofail Morocco, Kenitra, Morocco

Article Info

Article history:

Received Jun 24, 2023
Revised Aug 24, 2023
Accepted Dec 17, 2023

Keywords:

BABCN algorithm
Internet of things
Intrusion detection
Machine learning
Neural networks

ABSTRACT

The number of devices linked to the internet is rapidly increasing as the internet has become ingrained in every aspect of modern life. However, certain issues are getting worse, and their resolutions are not well-defined. One of the main issues is convergence and speed for communication between different internet of things (IoT) devices and their security. For that purpose, in this paper, an improved artificial bee colony (ABC) algorithm with binary search equations along with neural networks is proposed, known as the artificial bee colony algorithm with binary search equations (BABCN) algorithm for intrusion detection in terms of convergence and speed for communication. The depth-first search framework and binary search equations on which the artificial bee colony algorithm with binary search equations algorithm is built improve the algorithm's capacity for exploitation and speed up convergence. The initial weight and threshold value of the ABC neural networks are optimized using an algorithm to prevent them from entering a local optimum during the training procedure and accelerating training. The NSL-KDD dataset was used, and based on the results; the proposed algorithm improves classification and has high intrusion detection ability in the network. The proposed has undergone tests to be evaluated, and the results show that it performs better in detection accuracy, time, and false positive rate.

Corresponding Author:

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

1. INTRODUCTION

The number of devices associated with the internet is rapidly rising as the internet has become ingrained in every aspect of modern life. Particularly, internet of things (IoT) gadgets are becoming commonplace in everyday life. However, certain issues are becoming worse, and their solutions are also being discussed by different researchers [1]. In cloud and IoT security techniques, intrusion detection is used to identify, verify, and thwart illegal entry into a computer network or internetwork. Due to the impressive developments in data technology, there are important network confidentiality battles to win. Consequently, it is imperative to have an intrusion detection system (IDS) for the security of a network [2].

IDS fall under several categories of distinct approaches. The two primary divisions are active and inactive IDS. The traditional active IDS is unable to address newly emerging threats. Due to its enormous number of components and features, one of the primary challenges in finding intrusions is to locate and distinguish between regular and anomalous network connections. IDS is frequently used to determine how and where intrusions occur. The investigators conducted a thorough investigation of several element selection strategies to achieve real-time intrusion detection [3].

A compelling argument for improving the accuracy and speed of categorization schemes is to reduce the number of features based on the selection of the essential characteristics. Machine learning techniques have been widely used to recognize various attack types, and they can assist network administrators in responding to network attacks by guiding them toward the best course of action. The majority of these conventional machine learning techniques, however, fall within the shallow learning category and require extensive feature extraction and feature selection [4]. Due to its enormous number of components and features, one of the primary challenges in finding intrusions is to locate and distinguish between regular and anomalous network connections. IDS is frequently used to determine how and where intrusions occur. The classifier, which uses a detection mechanism to distinguish between intrusion and normal activity, is the fundamental component of an IDS. It can be difficult to implement a classifier with an accurate detection method, especially in IoT and cloud computing networks with lots of devices [5], [6]. Figure 1 presents the structure of IoT and cloud computing (CC) integration and working criteria. The rest of this paper is structured as follows: section 2 presents information about related work, section 3 discusses the proposed algorithm, section 4 covers the parameters, section 5 presents the results, and section 6 discusses the conclusion.

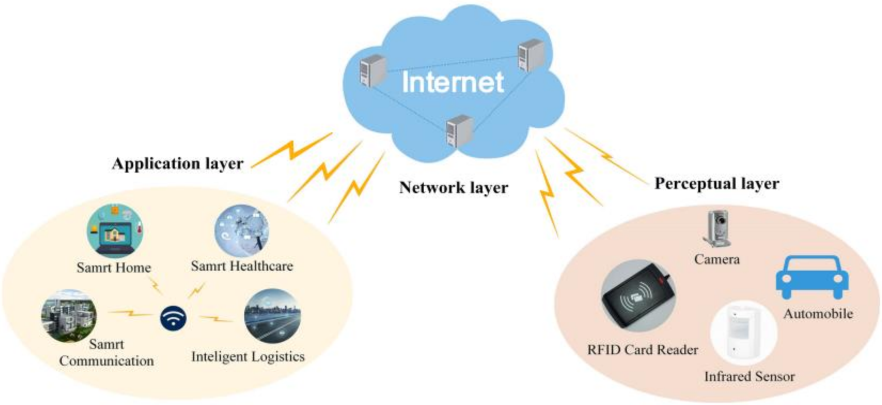


Figure 1. Structure of CC and IoT [7]

2. RELATED WORK

The two main types of intrusion detection techniques are anomaly-based and signature-based. With signature-based approaches, several intrusion patterns that have been tested and proven effective against them are stored in the system as predefined signatures. Additionally, the system compares the actions taken with these patterns, and if a similar pattern is seen, it will be labeled as an intrusion. Naturally, these techniques cannot identify brand-new or zero-day risks. These techniques, however, are particularly good at identifying recognized risks and their patterns [8]. A vision of typical activity is constructed using anomaly-based methods, after which an anomaly may denote an intrusion. It is well recognized that because there is no set pattern for monitoring, aberrant intrusions are exceedingly challenging to find. An occurrence is typically deemed abnormal if it occurs considerably more frequently or less frequently than a threshold [9]. Some AI methods employ tree-based algorithms like decision trees and random forests, which can build a structure for successfully detecting infiltration. In a decision tree algorithm, decisions are made step by step in accordance with the parameters of the problem. However, a decision tree may not always be sufficient to model a problem. Therefore, multiple decision trees are employed in random forest algorithms to improve overall decision-making accuracy. For software-defined networks, Xu *et al.* [10] have presented an anomaly-based method (IDSML) that enhances detection performance by combining many distinct tree-based methods. Neural networks are employed in other AI methods to accurately determine whether a specific occurrence resembles known patterns. Neural networks are made up of a number of interconnected nodes and have the ability to recognize patterns. According to Revathi and Malathi [11], calculations in a neural network take a long time since decision-making problems have a lot of parameters. Neural networks have been the primary detection method in numerous studies.

The artificial bee colony (ABC) algorithm was created in 2005 by Karaboga as a heuristic swarm intelligence system to resemble the group behavior of honeybees. It was initially created to address some issues with numerical optimization. According to Vinayakumar *et al.* [12], the ABC algorithm was used to optimize multivariate functions, and it was compared to other methods like the genetic algorithm (GA) and particle swarm optimization (PSO). The results show that ABC is a superior algorithm over others. On the other hand, the ABC algorithm struggles with exploitation and is prone to settle into a local optimum while excelling in exploring the answer. The GABC algorithm, which enhances exploitation by including information on the global optimal solution in the solution search equation, was introduced as an upgrade to the ABC algorithm [13]. According to Mishra *et al.* [14], a multi-strategy ensemble artificial bee colony (MEABC) algorithm was suggested. In MEABC, a variety of unique solution search tactics cohabit and compete for offspring throughout the search process. When applied to continuous optimization issues, the MEABC approach significantly enhances the performance of ABC. According to Karaboga and Ozturk [15], an ABC algorithm incorporating elite-guided search equations and a depth-first architecture, called DFSABC elite, was introduced. The algorithm's ability to be exploited is improved by giving superior solutions higher priority for computational resources.

2.1. Contribution of paper

The main purpose of the suggested model for the developed approach is outlined below NSL-KDD dataset. The network traffic signal is directly picked information in the NSL-KDD dataset. The IDS typical analysis is carried out by this dataset. Information preprocessing groups information for categorization and eliminates repetitive and unexpected occurrences. Selection of a feature it has been determined that the particular subset used the random feature selection method ABC and single feature selection method Gaussian distribution strategies to further the categorization. Hybrid categorization is to increase the accuracy of categorization, categorization is carried out using the artificial bee colony algorithm with binary search equations (BABCN) schemes [16].

3. PROPOSED ALGORITHM

The population-based, iterative ABC method is a powerful approach for tackling numerical optimization issues. The previous papers mentioned are [17]. Equations are stronger for exploration than for exploitation. Additionally, the ABC algorithm's convergence performance is not outstanding. Therefore, in [18], a binary search framework (BSF) and two search equation solutions, as given in (1), were suggested to better balance exploration and exploitation. This process, known as BSF, is used for improving the algorithm's ability to be exploited. The BSF framework can give better solutions higher priority when allocating more computational resources. The search equations retain the answer with the highest fitness value on each iteration, hastening the algorithm's training [19].

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

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

Where the solutions X_e and X_k were randomly selected from the binary search solution and the current population, respectively. Neither e nor k are equivalent to one another. X_{best} is currently the best choice. i, j and e, j are two random real values in the range [-1, 1]. In order to better balance ABC exploration and exploitation capacities, in paper [20], the problem that the candidate solution search equation in paper [21] has an overly significant disruption to the search solution is addressed. It then presents a binary search equation.

Different search equations should be utilized for the candidate solutions and the accepted solutions. Where X_k is a randomly chosen solution from the current population and X_e is a solution chosen at random from the binary search solution. e and k are not interchangeable terms. Right now, X_{best} is the best option. Two random real variables in the [-1, 1] in the given range. The issue that the candidate solution search equation in paper [22] has an excessively significant disruption to the search solution is addressed in article [23] in order to better balance ABC's exploration and exploitation capacities. A binary search equation is then presented. Different search equations should be employed for the candidate solutions and the accepted solutions.

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

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

In this situation, N stands for the Gaussian distribution, $gbest + pbest_i$ for the mean, and $gbest, pbest_i$ for the standard deviation. The Gaussian distribution in (3) is used to take advantage of the information around $pbest$ and $gbest$. According to (4) a comparable Gaussian search equation is suggested [24].

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

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

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

Finally, the neural network is trained using the backpropagation method using the initial weight and threshold values produced by the BABCN algorithm. By using gradient descent, the backpropagation method attempts to reduce the training error. The neural network for network traffic intrusion detection will employ the weights and thresholds with the minimum training error as its parameters [25]. The working criteria of the proposed backpropagation and neural network are as follows: choose a sample of data for training, then generate the weight values at random for the connections between the hidden layer neurons and the output layer neurons (ω_{jk}) and the hidden layer neurons and the input layer neurons (ω_j). Additionally, create the threshold values j of the neurons in the hidden layer and k of the output layer [26].

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

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

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

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

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

According to (8), the neural network's error is estimated. If the error fulfills the criteria, (9) and (10) are followed; otherwise, (11) is followed with (12).

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

In (13) and (14) modify the threshold and weight values between the hidden layer and the output layer. The weight and threshold values between the input layer and the hidden layer are changed in accordance with (15) [27], [28].

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

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

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

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

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

The 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, can be obtained after the results found in (16) [29]. After learning the results from (17) [30], it is possible to recover 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 $\omega_{jk}$and the new threshold values k between the hidden layer and the output layer.

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

Rerun into step (18) using the updated weight and threshold values. Stop the training process if the error complies with the specifications. Otherwise, obtain the relevant output signal from the neural network by using the present weights and thresholds as neural work input signals. The goal function of (18) is set to the loss function of a neural network, (19). Decide what the max cycle number (MCN) should be. Figure 2 presents the working approach [31].

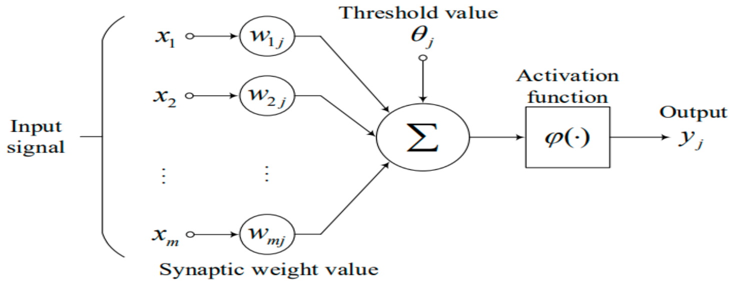


Figure 2. Working on the proposed approach

4. EVALUATION METRICS

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

$$AC = \frac{TP+TN}{TP+TN+FP+FN} \tag{20}$$

Accuracy (AC) is defined by (20) as the proportion of samples that have been correctly identified to all samples (41) [28].

$$TPR = \frac{TP}{TP+FN} \tag{21}$$

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) [32].

$$FPR = \frac{FP}{FP+TN} \tag{22}$$

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) [33].

4.1. Data set

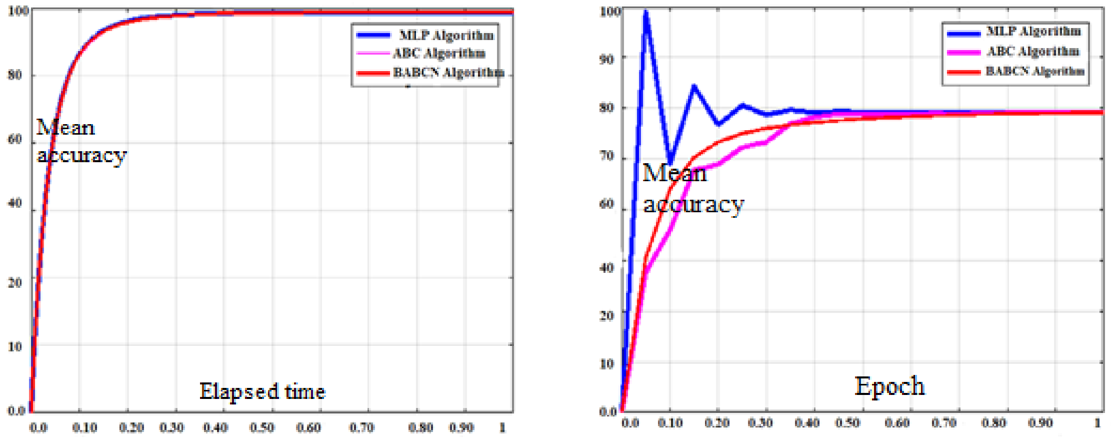
A data set called NSL-KDD has been proposed to address some of the underlying issues. The NSL-KDD train and test sets have a respectable number of records. Due to this benefit, all of the data can be used for the tests instead of just a tiny sample that must be chosen at random. As a result, evaluation findings from various research projects will be comparable and consistent [34].

5. RESULTS AND DISCUSSION

The results obtained from the multiclass classification are compared with the proposed algorithm in Table 1. Categorization issues arise when choosing a model's threshold. The two parameters of the receiver operating characteristic (ROC) curve, genuine positives and rate of false positives are mentioned in Table 1. In terms of deciding which data to employ for classification analysis, an area under the curve (AUC) is the best predictor of a model. The ROC curve is one instance of its use. The true positive rate in this case is compared against the false positive rate. Table 2 illustrates how the random forest (RF) performed well for multiclass classification as a whole.

The performance measurement tool is displayed in an ROC curve. Categorization issues arise when choosing a model's threshold. The two parameters of this ROC curve are genuine positives and the rate of false positives. Table 2 displays the outcomes of a 32-batch operation. In this case, the mean accuracy of the proposed BABCN algorithm classifier declined as the number of research epochs increased. When the number of epochs increased from 10 to 32, the accuracy decreased. Figure 3 presents the batch operation of different algorithms, Figure 3(a) shows the elapsed time of the different algorithms, and Figure 3(b) shows the epoch time of the different algorithms.

Tables 3 and 4 display the outcomes for batch sizes of 64 and 128. The mean accuracy of the proposed BABCN algorithm classifier seemed to have increased as the number of research epochs grew. When the number of epochs increased from 15 to 45, there was a minor decrease for the BABCN, and then it increased at 45 epochs. Table 4 shows that a larger batch size could result in a shorter duration time. Figure 4 presents the accuracy of different approaches.



(a) (b)

Figure 3. The batch operation of (a) elapsed time of different algorithms and (b) epoch time of different algorithms

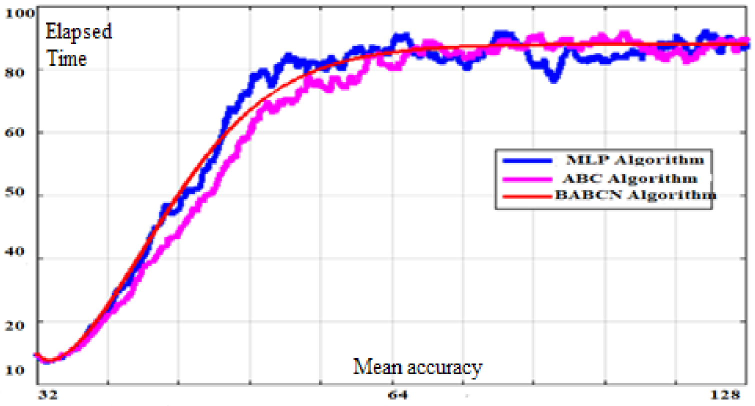


Figure 4. Accuracy of different approaches

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	
Proposed BABCN algorithm	0.98	0.98	0.98	1	0.95

Multilayer perceptron (MLP); denial of service (DoS); distributed denial of service (DDoS)

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
Proposed BABCN algorithm	32	0.99	0.97

Table 3. Metrics batch size 64

Algorithm	Epoch	Mean accuracy	Elapsed time
MLP algorithm	25	0.98	0.99
ABC algorithm	35	0.97.6	0.99
Proposed BABCN algorithm	45	0.98.8	0.99

Table 4. Metrics batch size 128

Algorithm	Epoch	Mean accuracy	Elapsed time
MLP algorithm	35	0.98	0.99
ABC algorithm	40	0.98	0.99
Proposed BABCN algorithm	49	0.99	0.99

6. CONCLUSION

We looked at various machine learning and deep learning techniques on an IoT network and compared them with our proposed approach in this study. We took into account the analysis of RF, convolutional neural network (CNN), MLP, and the proposed BABCN algorithm. The best outcome in terms of multiclass classification accuracy and AUC was achieved by random forests and CNN. In trials with 32 and 64 batches, the accuracy slightly decreased with the addition of epochs, whereas in trials with 128 batches, the accuracy slightly increased. Additionally, we discovered that boosting the batch size helped hasten the computation. For the proposed BABCN algorithm, increasing the batch size by two could speed up computation by 1.3–2.4 times, while for CNN, it could accelerate computation by 1.8–2.4 times. Our long-term objective is to create models using the proposed BABCN algorithm. Future deployment of our proposed system aims to deliver detection and classification services against various cyber-attacks and intrusions within a network of IoT devices (e.g., a network of advanced RISC machines (ARM) or Arduino Raspberry Pi nodes).

ACKNOWLEDGMENTS

Thanks to Ibn-Tofail University Kenitra, Morocco for their support and My SV for his support and time.

REFERENCES

[1] M. S. Noori, R. K. Z. Sahbudin, A. Sali, and F. Hashim, “Feature drift aware for intrusion detection system using developed variable length particle swarm optimization in data stream,” *IEEE Access*, vol. 11, pp. 1–1, 2023, doi: 10.1109/access.2023.3333000.

[2] H. Gupta, S. Sharma, and S. Agrawal, “Artificial intelligence-based anomalies detection scheme for identifying cyber threat on iot-based transport network,” *IEEE Transactions on Consumer Electronics*, pp. 1–1, 2023, doi: 10.1109/tce.2023.3329253.

[3] F. Feng, K. C. Li, J. Shen, Q. Zhou, and X. Yang, “Using cost-sensitive learning and feature selection algorithms to improve the performance of imbalanced classification,” *IEEE Access*, vol. 8, pp. 69979–69996, 2020, doi: 10.1109/ACCESS.2020.2987364.

[4] W. A. H. M. Ghanem *et al.*, “Cyber intrusion detection system based on a multiobjective binary bat algorithm for feature selection and enhanced bat algorithm for parameter optimization in neural networks,” *IEEE Access*, vol. 10, pp. 76318–76339, 2022, doi: 10.1109/ACCESS.2022.3192472.

[5] Y. Gong, Y. Fang, L. Liu, and J. Li, “Multi-agent intrusion detection system using feature selection approach,” in *Proceedings - 2014 10th International Conference on Intelligent Information Hiding and Multimedia Signal Processing, IIH-MSP 2014*, IEEE, Aug. 2014, pp. 528–531. doi: 10.1109/IIH-MSP.2014.137.

[6] L. Hakim, R. Fatma, and Novriandi, “Influence analysis of feature selection to network intrusion detection system performance using NSL-KDD dataset,” in *Proceedings - 2019 International Conference on Computer Science, Information Technology, and Electrical Engineering, ICOMITEE 2019*, IEEE, Oct. 2019, pp. 217–220. doi: 10.1109/ICOMITEE.2019.8920961.

[7] Y. Su, K. Qi, C. Di, Y. Ma, and S. Li, “Learning automata based feature selection for network traffic intrusion detection,” in *Proceedings - 2018 IEEE 3rd International Conference on Data Science in Cyberspace, DSC 2018*, IEEE, Jun. 2018, pp. 622–627. doi: 10.1109/DSC.2018.00099.

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

[9] X. Zhang, P. Zhu, J. Tian, and J. Zhang, “An effective semi-supervised model for intrusion detection using feature selection based LapSVM,” in *IEEE CITS 2017 - 2017 International Conference on Computer, Information and Telecommunication Systems*,

IEEE, Jul. 2017, pp. 283–286. doi: 10.1109/CITS.2017.8035323.

[10] H. Xu, Y. Fu, C. Fang, Q. Cao, J. Su, and S. Wei, "An improved binary whale optimization algorithm for feature selection of network intrusion detection," in *Proceedings of the 2018 IEEE 4th International Symposium on Wireless Systems within the International Conferences on Intelligent Data Acquisition and Advanced Computing Systems, IDAACS-SWS 2018*, IEEE, Sep. 2018, pp. 10–15. doi: 10.1109/IDAACS-SWS.2018.8525539.

[11] S. Revathi and A. Malathi, "A detailed analysis on NSL-KDD dataset using various machine learning," *International Journal of Engineering Research & Technology*, vol. 2, no. 12, pp. 1848–1853, 2013.

[12] 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.

[13] 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, 2020, doi: 10.1007/s00500-019-04030-2.

[14] 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 and Tutorials*, vol. 21, no. 1, pp. 686–728, 2019, doi: 10.1109/COMST.2018.2847722.

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

[16] 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, 2014, doi: 10.1016/j.engappai.2014.07.012.

[17] W. Zou, Y. Zhu, H. Chen, and X. Sui, "A clustering approach using cooperative artificial bee colony algorithm," *Discrete Dynamics in Nature and Society*, pp. 1–17, 2010.

[18] T. Alam, R. Gupta, S. Qamar, and A. Ullah, "Recent applications of artificial intelligence for sustainable development in smart cities," *Studies in Computational Intelligence*, vol. 1061, pp. 135–154, 2022, doi: 10.1007/978-3-031-14748-7_8.

[19] 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, 2014, doi: 10.1007/s00521-014-1685-y.

[20] 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, 2012, doi: 10.1007/s10845-010-0393-4.

[21] A. Ullah, N. M. Nawi, H. Bin Mahdin, S. Baseer, and M. M. Deris, "Role of different integer virtual machine in cloud data centre," *International Journal on Informatics Visualization*, vol. 3, no. 4, pp. 394–398, 2019, doi: 10.30630/joiv.3.4.274.

[22] 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, 2023, doi: 10.1007/s11760-022-02231-1.

[23] 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, 2019, doi: 10.1109/TII.2018.2881543.

[24] 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, 1986, doi: 10.1016/0169-2607(86)90106-9.

[25] J. Veitch *et al.*, "Parameter estimation for compact binaries with ground-based gravitational-wave observations using the LALInference software library," *Physical Review D - Particles, Fields, Gravitation and Cosmology*, vol. 91, no. 4, pp. 1–26, 2015, doi: 10.1103/PhysRevD.91.042003.

[26] 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, pp. 62–71, 2023, doi: 10.11591/ijict.v12i1.pp62-71.

[27] U. Ullah and A. Ullah, "An evolutionary algorithm for the solution of multi-objective optimization problem," *International Journal of Advances in Applied Sciences*, vol. 11, no. 4, pp. 287–295, 2022, doi: 10.11591/ijaas.v11.i4.pp287-295.

[28] A. Ullah, H. Aznaoui, C. B. Şahin, M. Sadie, O. B. Dinler, and L. Imane, "Cloud computing and 5G challenges and open issues," *International Journal of Advances in Applied Sciences*, vol. 11, no. 3, pp. 187–193, 2022, doi: 10.11591/ijaas.v11.i3.pp187-193.

[29] A. Ullah, S. A. Khan, T. Alam, S. Luma-Osmani, and M. Sadie, "Heart disease classification using various heuristic algorithms," *International Journal of Advances in Applied Sciences*, vol. 11, no. 2, pp. 158–167, 2022, doi: 10.11591/ijaas.v11.i2.pp158-167.

[30] S. Rai, A. Ullah, W. L. Kuan, and R. Mustafa, "An enhanced compression method for medical images using SPIHT Encoder for fog computing," *International Journal of Image and Graphics*, 2023, doi: 10.1142/S0219467825500251.

[31] S. Ouhame, Y. Hadi, and A. Ullah, "An efficient forecasting approach for resource utilization in cloud data center using CNN-LSTM model," *Neural Computing and Applications*, vol. 33, no. 16, pp. 10043–10055, 2021, doi: 10.1007/s00521-021-05770-9.

[32] S. Ouhame, Y. Hadi, and 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*, vol. 16, no. 14, pp. 4–17, 2020, doi: 10.3991/ijoe.v16i14.16623.

[33] S. Umar, S. Baseer, and Arifullah, "Perception of cloud computing in universities of Peshawar, Pakistan," in *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*, IEEE, Aug. 2017, pp. 87–91. doi: 10.1109/INTECH.2016.7845046.

[34] Arifullah, S. Baseer, and S. Umar, "Role of cooperation in energy minimization in visual sensor network," in *2016 6th International Conference on Innovative Computing Technology, INTECH 2016*, IEEE, Aug. 2017, pp. 447–452. doi: 10.1109/INTECH.2016.7845026.

BIOGRAPHIES OF AUTHORS

Imane Laassar working as a research assistant at the 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 a 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.