Computers, Materials & Continua
DOI: 10.32604/cmc.2022.025204
Article

Tech Science Press

Artificial Intelligence Based Data Offloading Technique for Secure MEC Systems

Fadwa Alrowais[1], Ahmed S. Almasoud[2], Radwa Marzouk[3], Fahd N. Al-Wesabi[4,5], Anwer Mustafa Hilal[6,*], Mohammed Rizwanullah[6], Abdelwahed Motwakel[6] and Ishfaq Yaseen[6]

[1]Department of Computer Sciences, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[2]Department of Information Systems, College of Computer and Information Sciences, Prince Sultan University, Rafha Street, Riyadh, 11586, Saudi Arabia
[3]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[4]Department of Computer Science, College of Science & Arts at Mahayil, King Khalid University, Muhayel Aseer, 62529, Saudi Arabia
[5]Department of Information Systems, Faculty of Computer and Information Technology, Sana'a University, 61101, Yemen
[6]Department of Computer and Self Development, Preparatory Year Deanship, Prince Sattam bin Abdulaziz University, Al-Kharj, 16278, Saudi Arabia
*Corresponding Author: Anwer Mustafa Hilal. Email: a.hilal@psau.edu.sa
Received: 16 November 2021; Accepted: 06 January 2022

Abstract: Mobile edge computing (MEC) provides effective cloud services and functionality at the edge device, to improve the quality of service (QoS) of end users by offloading the high computation tasks. Currently, the introduction of deep learning (DL) and hardware technologies paves a method in detecting the current traffic status, data offloading, and cyberattacks in MEC. This study introduces an artificial intelligence with metaheuristic based data offloading technique for Secure MEC (AIMDO-SMEC) systems. The proposed AIMDO-SMEC technique incorporates an effective traffic prediction module using Siamese Neural Networks (SNN) to determine the traffic status in the MEC system. Also, an adaptive sampling cross entropy (ASCE) technique is utilized for data offloading in MEC systems. Moreover, the modified salp swarm algorithm (MSSA) with extreme gradient boosting (XGBoost) technique was implemented to identification and classification of cyberattack that exist in the MEC systems. For examining the enhanced outcomes of the AIMDO-SMEC technique, a comprehensive experimental analysis is carried out and the results demonstrated the enhanced outcomes of the AIMDO-SMEC technique with the minimal completion time of tasks (CTT) of 0.680.

Keywords: Data offloading; mobile edge computing; security; machine learning; artificial intelligence; XGBoost; salp swarm algorithm

1 Introduction

Internet of Things (IoT) has heterogeneous resource-based environment which is applied for providing continuous service for fulfilling the end-user requirement. The idea of IoT varies from small sensor which comprises processing, and communication features with storage space. The intelligence of this human communicative behaviour and machines are used for assisting user requirements [1]. With the wide-ranging application of communication tools and several models, the IoT has been employed in the military services, medical sector, mobile communications, industrial fields, etc. The number of information responsible to IoT could not be determined particularly due to the density of a device, the random memory space, and request rate [2]. Mobile Edge Computing (MEC) is an alternative transmission method which provides service to devoted users in the transmitting network. The derived resource is extended to user for offering complex services. The facilities such as mobility, support transformation, elasticity, reliability, and adaptability features support different user device densities. Fig. 1 depicts the framework of MEC [3].

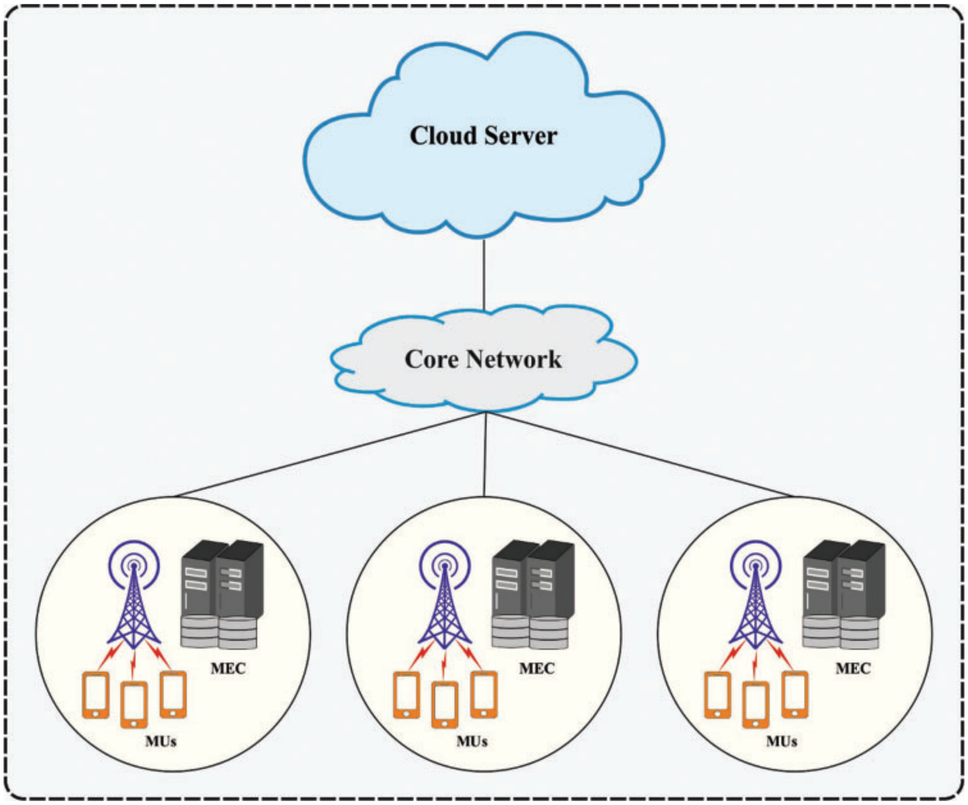

Figure 1: MEC structure

MEC has technologically advanced as a considerable solution for providing nearby distance to the mobile end-users and computing resource at the network edge [4]. The main advantage of MEC technology comprises: reduce the energy consumption of mobile gadgets by mitigating the problem of execution its computing task, giving location awareness, offering exact computing results in an appropriate way, improving the efficacy of the mobile applications, and it is possible to reduce the latency. However, the adoption of MEC method from the whole network architecture was developed for network congestion, MEC server computation capacity, and end-user QoS condition [5].

The challenges of data offloading in the end-users to MEC for further computing are extensively studied from present research while examining the computational and transmission limitations [6].

Various methods and techniques to computational off-loading in MEC were developed from the study with the purpose of assigning radio resources effectively, energy utilization, and decreasing the computational latency. Achieving the best possible offloading result in complex and dynamic multi-user wireless MEC systems are complex process. Additionally, the security threats confronted during data forwarding have not been addressed in each offloading method [7]. Moreover, insufficient data security checks might quickly exceed the advantage of MEC method. In order to counter the cyber threats in MCC, it is crucial to detect early cyber threats, therefore performing fast countermeasures to prevent the risks [8].

Currently, it has various techniques presented for preventing and detecting cyber risks in cloud platforms. The usual limitation of this technique is relatively minimal accuracy from detect cyber threat, and they could not work effectively in cloud system [9]. In this case, they presented a framework utilizing an advanced detecting methodology developed from the DL method that permits detection of various threats with high precision. The DL technique is a subdivision of ML method which emphasizes the approach inspired by the structure and function of NN. In recent times, DL method was performed efficiently in various domains.

Hilal et al. [10] develop a powerful DL based data offloading and cyberattack detection (DL-DOCAD) technology for MEC. The aim of the presented method is to improve the QoE in MEC system. The presented model contains attack detection, traffic prediction, data offloading and. This method employs a GRU based prediction methodology to traffic detection. Additionally, the BSA-FFNN technique is employed as a detector for cyber threats in MEC. Elgendy et al. [11], presented a DL methodology for detecting the best possible solution.

In Mitsis et al. [12], the combined problems of MEC server selection by end-user and the optimum price setting, and data offloading by the MEC are examined in several end-user environments and MEC servers. The programmability and flexibility provided by the SDN technique allow the real time execution of the presented architecture. At first, an SDN controller performs an RL architecture based stochastic learning automata to enable the end-users for selecting MEC servers for data offloading.

Alkatheiri [13] presents privacy-controlled offloading system to improve the data sharing security among the edge-IoT gadgets. In this system, harmonized trust validation with recursive decision making is implemented, afterward the data offloading method. Xu et al. [14], proposed a time-effective offloading technique (TEO) with privacy preservation to smart sensors in edge computing. In fact, the offloading and time utilization of secured information is examined in a formal manner. Next, an enhanced Strength Pareto Evolutionary Algorithm (SPEA2) is leveraged for jointly optimizing the average privacy entropy and time utilization. Eventually, various experiment assessments were performed for verifying the reliability and efficiency of this approach.

This study introduces an artificial intelligence with metaheuristic based data offloading technique for Secure MEC (AIMDO-SMEC) systems. The proposed AIMDO-SMEC technique incorporates an effective traffic prediction module using Siamese Neural Networks (SNN) to determine the traffic status in the MEC system. Also, an adaptive sampling cross entropy (ASCE) technique is utilized for data offloading in MEC systems. Moreover, the modified salp swarm algorithm (MSSA) with extreme gradient boosting (XGBoost) technique is applied for the identification and classification of cyberattacks that exist in the MEC systems. For examining the enhanced outcomes of the AIMDO-SMEC technique, a comprehensive experimental analysis is carried out and the results demonstrated the enhanced outcomes of the AIMDO-SMEC technique interms of different measures.

The rest of the paper is organized as follows. Section 2 introduces the proposed AIMDO-SMEC technique. Then, Section 3 validates the proposed model and Section 4 concludes the study.

2 The Proposed Model

In this study, a novel AIMDO-SMEC technique has been developed to accomplish effective data offloading and security in MEC. The projected AIMDO-SMEC approach encompasses three major stages namely SNN based traffic flow prediction, ASCE based data offloading, and MSSA with XGBoost based cyberattack classification.

2.1 Design of SNN Based Traffic Prediction Model

Initially, the SNN model receives the input data and then effectively determines the traffic status in the MEC systems. SNN contains identical networks that accept dissimilar inputs however, they are combined with an energy function at the top. This process calculates few metrics among the high-level feature depiction on all the sides. The variables among the identical networks are tied. Weight tying guarantees that 2 absolutely equal images couldn't be mapped by their corresponding network to distinct positions in feature space since all the networks compute the same functionalities. As well, the network is symmetric, therefore we proposed 2 different images to the identical networks, and the topmost conjoining layers would evaluate the similar metrics since we propose the similar 2 images to the opposite twins. The conventional method is a Siamese convolution neural network with L layer with N_l unit, in which $h_{1,l}$ denotes the hidden layer in l for the initial twin, and $h_{2,l}$ represent the second twin. They utilize exclusive rectified linear (ReLU) units in the initial $L-2$ layer and sigmoidal units in the residual layer.

The algorithm contains a series of convolution layers, all of which utilize an individual with a filter of differing size and a fixed stride of 1. The amount of convolution filters is stated as a multiple of 16 to enhance the performances. Therefore, the kth filter map in all the layers take the succeeding form:

$$a_{1,m}^{(k)} = \max\ -pool(\max\,(O,\ W_{l-1,l}^{(k)} \star h_{1,(l-1)} + b_l),\ 2) \tag{1}$$

$$a_{2,m}^{(k)} = \max - pool(\max\,(O,\ W_{l-1,l}^{(k)} \star h_{2,(l-1)} + b_l),\ 2) \tag{2}$$

whereas $W_{l-1,l}$ represent the 3D tensor represent the feature maps for l layer and take $\star$ as valid convolution function corresponds to returning only those output units that are results of comprehensive overlaps among the input feature maps and every convolution filter. The unit in the last convolution layer is flattened into an individual vector. This convolution layer afterward an FC layer, additional layer computing the induced distance metrics among all the Siamese twins, i.e., provided to an individual sigmoidal output unit [15]. Accurately, the prediction vector is provided as $p = \sigma(\sum_j \alpha_j |h_{1,L-1}^{(j)} - h_{2,L-1}^{(j)}$ whereas σ represent the sigmoidal activation function. This last layer induces metrics on the learned feature space of the $(L-1)th$ hidden neuron and scores the similarities among the 2 feature vectors. The α_j are further variables which are learned using the model at the time of training, weighing the significance of the element-wise distance. This determines a last Lth FC layer for the network that links 2 Siamese twins.

2.2 Design of ASCE Based Data Offloading Process

During data offloading process, the ASCE manner was utilized to offload the data in the MEC systems. The CE is termed as probability concept as Kullback Leibler (KL) deviation, also it is simplified as measure of distance between 2 probability distributions. In the event of 2 distributions, $q(x)$ & $p(x)$, CE can be determined as follows

$$D(q||p) = \underbrace{\sum^{q}(x)lnq(x)}_{H(q)} - \underbrace{\sum^{q}(x)lnp(x)}_{H(q,p)} \tag{3}$$

During the representation of CE, is used to cost function in ML, and solves the problem using probability learning. It is learn in $p(x)$ that is generally a trained sample is generated for attaining an optimal policy of X based $p(x)$ is equivalent for empirical solutions, $q(x)$. During the case of probabilities learning, probability distribution function $p(x)$ is used by the descriptor u, $p(x,\ u)$ represents a Gaussian distribution, also u is made up of variance and mean. The CE in Eq. (3) used as lost function. It denotes that lesser $H(q,p)$ is, lesser the distance in $q(x)$ & $p(x)$.

$$\min H(q,p) = \max \sum q(x)\ln p(x) = \max \frac{1}{S}\sum \ln p(x,\ u) \tag{4}$$

Whereas $q(x)$ is $\frac{1}{S}$, the probability of autonomous outcome in the group of samples $1/S$ in which S represents the cardinality of group [16]. According to the problems in Eq. (4), it is exploited to recognize the optimum indicator u to minimize $H(q,p)$. In tth iteration, S series of arbitrary samples x, is allotted as a candidate i.e., recovered according to the probability $p(x,\ u)$. The probable instance is made by adoptive sampling has been determined. It is estimated that the $\{\Psi(x^{s})\}_{s=1}^{S}$ objectives are estimated and organizing as $\Psi(x^{[1]}) \leq \Psi(x^{[2]}) \leq \cdots \leq \Psi(x^{[S]})$. Next S_{elite} samples, like, $x^{[1]}, x^{[2]}, \ldots, x^{[elite]}$, offer lower objective, is elective as elites. Next, u optimum indicator for x approach is evaluated as:

$$u^{*} = \arg\ \max_{\mathrm{u}} \frac{1}{S} \sum_{s=1}^{S_{e1ite}} \ln p(x^{[s]},\ u) \tag{5}$$

Using (3) & (5) and encourage $\dfrac{\partial H(q,p)}{\partial u_{l}} = 0$, the saddle point c

$$u_{l}^{*} = \frac{1}{S_{e1ite}} \sum_{s=1}^{S_{e1ite}} x_{l}^{[s]} \tag{6}$$

In the projected algorithm, the CE interrelated measures are employed for extending the probability. In supposing the arbitrariness of sampling, count of instances is lesser, the $u^{(t+1)}$ function is updated in $(t+1)th$ iteration according to u^{*} which is handled by (5) & (6), and $u^{(t)}$ is learned from the previous iteration.

$$u^{(t+1)} = \alpha u^{*} + (1-\alpha)u^{(t)} \tag{7}$$

Whereas $\alpha \in [0,\ 1]$ signifies the learning values. In general, for CE assisted component the round converges a better solution to the issue.

2.3 Design of MSSA with XGBoost Based Cyberattack Detection Model

At the final stage, the XGBoost model is utilized to identify the occurrence of cyberattacks. XGBoost is an ensemble-based classification method and was developed by Chen et al. [17]. XGBoost employs boosted tree and is employed for regression and classification. XGBoost was extensively utilized for several predictive tasks and produce considerable results because of effective learning speed and capability. XGBoost is an improved form of the gradient boosting tree. The key objective is to decrease the computation resource consumption, loss, and model complexity. The difficulty can be decreased by regularization. Furthermore, the method normalization is employed for alleviating the over-fitting concept. The purpose of utilizing XGBoost for medical data is because of its innate ability to manage the data imbalance. This process operates by including the trees repeatedly by partitioning the feature. In each iteration, the loss decreases, and novel rules are further. The process repeats till the algorithm attained effective performance. Consider D represent the dataset consisting of n number of features:

$$D = \{x_1, x_2, \ldots, x_n\} \tag{8}$$

Y signifies the class attribute; Y_i denotes the actual value, whereas Y_t characterizes the predictable value [18].

$$Tree_{Ens} = \sum_{k=1}^{j} Loss\left(y_k, \sum_{n=1}^{N} f_n(x_k)\right) + \sum_{n=1}^{N} \Omega(f_n), f_n \in F \tag{9}$$

In which Tree_Ens implies a tree ensemble method. *e loss indicates loss function that is variance among the actual and predicted. N signifies the amount of trees. F characterizes the set of trees utilized from the training model. Ω indicates the regularization term.

The parameters involved in the XGBoost model are optimally chosen by the use of MSSA. SSA is newly developed population-based meta-heuristic approach [19]. The main concept was invented from scavenging and propulsion events of salp inside ocean. Salps belong to the salpaide family with transparent and barrel shaped bodies. Inside Deep Ocean individual salps related to one another and make a chain as structure named salp swarm/salp chain at the time of food search. The entire population in the salp swarm could be assumed as, supporters and front-runners. Supporter salp alters their corresponding locations according to each other. The front-runner led the sequence from the anterior to the target by changing its own position as per the nourishment parameter when examining searching space. Mathematical position of front-runner salps are changed as

$$A_k^1 = \begin{cases} F_k + E_1((UB_k - LB_k)E_2 + LB_k)E_3 \geq 0 \\ F_k - E_1((UB_k - LB_k)E_2 + LB_k)E_3 \geq 0 \end{cases} \tag{10}$$

Whereas A_k^1 signifies location of leader in k^{th} parameter, F_k means location of food source in k^{th} parameter, UB_k signifies upper limit in k^{th} variables, LB_k symbolizes lower limit in k^{th} parameter, and E_1 indicates the exploration and exploitation factor (controlling variable)

$$E_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \tag{11}$$

In which l indicates existing iteration and L represents maximal amount of iteration. E_2 and E_3, denotes 2 arbitrary measures in among zero and one. Supporter salps position can be changed as follows

$$a_k^p = \frac{1}{2}(a_k^p + a_k^{p-1}) \tag{12}$$

Let, $p \geq 2$

$a_k^p = p^{th}$ follower salps location in k^{th} parameter.

$a_k^{p-1} = (p-1)^{th}$ supporter salps position in k^{th} dimension.

From the optimization perspective, SSA is considered as follows: the chain or swarm of salps is the searching agent and the deep salty ocean is the searching space. The leader salps discover and exploit the whole searching area by changing corresponding position as per the nourishment and save outcomes of all the iterations. Afterward accomplishment of each iteration global optimal solution has last optimal saved solution [20]. The SSA displays some benefits over other meta-heuristic approaches like (i) the method employs only one variable as controlling factor El i.e., linearly reduced (ii) it saves the optimum outcome observed in all iterations by utilizing the food source parameter for later work, once the population breaks down (i) the leader explores the searching area by changing the corresponding position by nourishment resource and the follower alters their corresponding position according to each other and move to the leader salps, that results in barring the model to fall in local optimal stagnation. Fig. 2 demonstrates the flowchart of SSA.

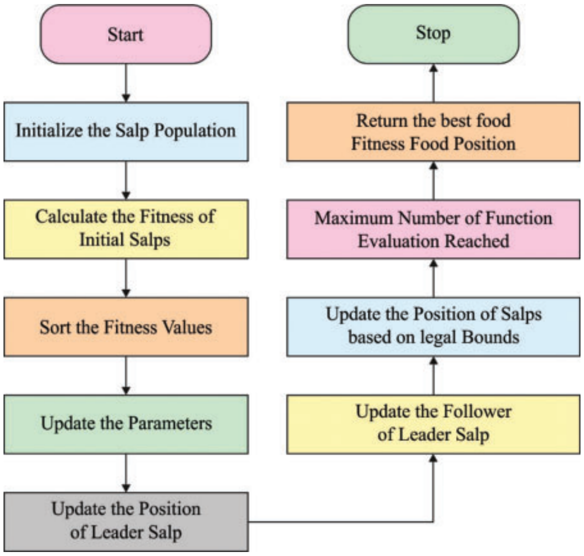
Start
Initialize the Salp Population
Calculate the Fitness of Initial Salps
Sort the Fitness Values
Update the Parameters
Update the Position of Leader Salp
Update the Follower of Leader Salp
Update the Position of Salps based on legal Bounds
Maximum Number of Function Evaluation Reached
Return the best food Fitness Food Position
Stop

Figure 2: Flowchart of SSA

Every nature simulated meta-heuristic technique selects the primary parameter arbitrarily as the candidate solution. Usually, the arbitrary population elective procedure follows uniform distribution method. As the learning techniques follows black box manner and could not need some background data it begins enhancing the candidate solution for getting optimal result still existing conditions are not attained. The efficiency of such techniques has restricted the count of time utilized by validating all candidate solutions inside the explore space for reaching neighboring optimum solutions as global optimal. In order to obtain global optimal, all adjacent optimum solutions have that scrutinized. Therefore, detection of neighboring optimum solution as only continued with primary guess of parameter could not sufficient in the explore space. Besides, for increasing the exploration ability from the explore space, when it can be assumed that opposite case concurrently with primary guess, afterward the possibility of obtaining neighboring optimum solutions near global optimal improves. For enhancing the exploration ability as well as avoiding early convergence from local minimal, novel evolutionary approach appeared, planned as space transformation search (STS). Based on STS process the ongoing search area was reallocated to fresher area for evaluating candidate solutions for both spaces concurrently. An STS approach includes 4 methods amongst that OBL is among them. The OBL method is written as $X \in [P, Q]$, where X refers to the real number and P, Q signifies the boundary values of exploring area. Afterward, due to OBL method opposite is represented as:

$$X^* = P + Q - X \tag{13}$$

where, $X^* \in [P + Q]$.

The fitness has computed then getting a novel place by utilizing OBL approach in the transmitted space. Afterward, the fitness has been related to the preceding observed fitness created in the primary population of SSA. Afterward, comparative further neighboring optimum solutions are assumed that next generation and continue the round still determined conditions not obtained.

3 Performance Validation

In this section, the experimental results analysis of the AIMDO-SMEC technique takes place and the results are reviewed under several aspects.

Tab. 1 and Fig. 3 showcases the N_{MSE} analysis of the SNN technique to examine the traffic prediction outcomes. The results ensured that the SNN technique has gained improved prediction outcomes with minimal values of N_MSE. For instance, with area of 5 sq. km, the SNN technique has obtained lower N_MSE values of 0.00962 but the LSTM, BiLSTM, and GRU models have provided higher N_MSE of 0.01348, 0.01105, and 0.00981. At the same time, with area of 15 sq. km, the SNN system has attained decreased N_MSE of 0.00201 but the LSTM, BiLSTM, and GRU manners have provided increased N_MSE values of 0.00416, 0.00275, and 0.00218 correspondingly. Likewise, with area of 25 sq. km, the SNN system has reached reduced N_MSE of 0.00036 while the LSTM, BiLSTM, and GRU manners have reached higher N_MSE of 0.00199, 0.00086, and 0.00062 correspondingly. Similarly, with area of 40 sq. km, the SNN method has achieved minimal N_MSE of 0.00007 although the LSTM, BiLSTM, and GRU manners have reached increased N_MSE values of 0.00073, 0.00027, and 0.00010 correspondingly.

Table 1: Predictive results analysis of SNN model

Area (Sq. km)	LSTM	BiLSTM	GRU	SNN
5	0.01348	0.01105	0.00981	0.00962
10	0.00615	0.00449	0.00320	0.00281
15	0.00416	0.00275	0.00218	0.00201
20	0.00360	0.00186	0.00159	0.00112
25	0.00199	0.00086	0.00062	0.00036
30	0.00084	0.00037	0.00027	0.00017
35	0.00078	0.00023	0.00010	0.00009
40	0.00073	0.00027	0.00010	0.00007

Tab. 2 and Fig. 4 depict the CTT analysis of the ASCE manner to study the traffic prediction outcomes. The results ensured that the ASCE method has reached increased prediction outcomes with lower values of CTT. For instance, with task count of 1, the ASCE system has obtained lower CTT of 0.424 but the Adaptive and fixed approaches have attained higher CTT of 0.675 and 0.651 correspondingly. Also, with task count of 5, the ASCE algorithm has obtained lower CTT of 0.513 while the Adaptive and fixed systems have obtained maximal CTT of 1.154 and 1.251 respectively. Similarly, with task count of 15, the ASCE approach has obtained lower CTT of 0.837 while the Adaptive and fixed systems have attained higher CTT of 2.176 and 2.436 correspondingly. At last, with task count of 20, the ASCE methodology has obtained minimal CTT of 0.983 but the Adaptive and fixed methodologies have achieved higher CTT of 2.971 and 3.198 respectively.

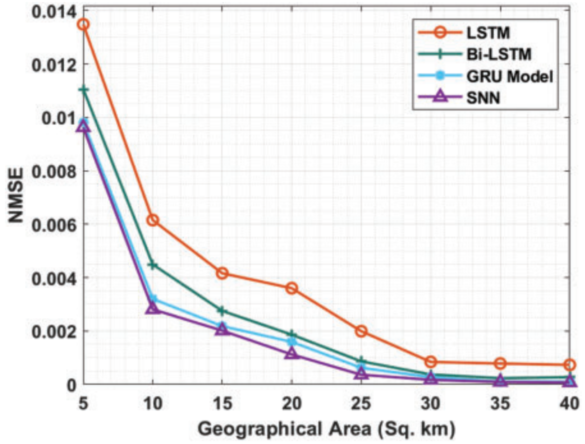

Figure 3: N_MSE analysis of SNN model

Table 2: Completion time of tasks (CTT) analysis of ASCE model

No. of tasks	Adaptive	Fixed	Proposed ASCE
1	0.675	0.651	0.424
2	0.943	0.845	0.432
3	0.975	1.016	0.456
4	1.089	1.097	0.472
5	1.154	1.251	0.513
6	1.227	1.284	0.529
7	1.292	1.421	0.561
8	1.405	1.567	0.586
9	1.397	1.576	0.610
10	1.559	1.746	0.626
11	1.567	1.811	0.659
12	1.714	1.933	0.716
13	2.046	2.208	0.805
14	1.981	2.200	0.805
15	2.176	2.436	0.837
16	2.290	2.509	0.862
17	2.403	2.622	0.870
18	2.598	2.833	0.894
19	2.898	3.141	0.951
20	2.971	3.198	0.983
Average	1.718	1.867	0.680

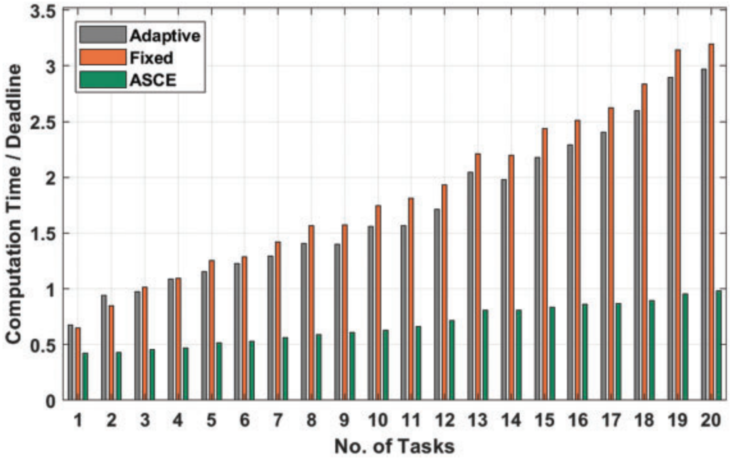

Figure 4: CTT analysis of ASCE technique with different count of tasks

A comprehensive results analysis of the MSSA-XBoost technique takes place under TS of 60% in Fig. 5. The results show that the RF technique has accomplished least performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.924, 0.938, 0.930, 0.936, and 0.860 respectively. At the same time, the DBN model has gained certainly increased outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.915, 0.958, 0.962, 0.916, and 0.913 respectively. Moreover, the optimal DBN model has attained moderate performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.990, 0.962, 0.977, 0.978, and 0.953 respectively. Furthermore, the optimal FFNN model has resulted in competitive outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.994, 0.975, 0.981, 0.986, and 0.969 respectively. However, the MSSA-XGBoost technique has portrayed the other methods with the maximal sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.998, 0.985, 0.9940, 0.996, and 0.981 respectively.

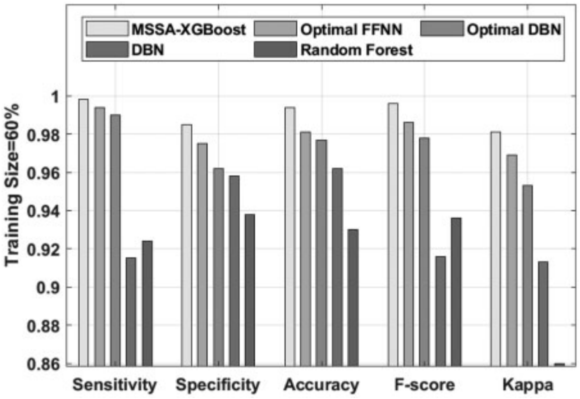

Figure 5: Comparative analysis of MSSA-XBoost technique under TS = 60%

A brief results analysis of the MSSA-XBoost approach take place under TS of 70% in Fig. 6. The outcomes demonstrated that the RF manner has accomplished worse performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.938, 0.945, 0.938, 0.944, and 0.892 correspondingly. Simultaneously, the DBN approach has reached certainly improved outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.925, 0.963, 0.968, 0.926, and 0.927 correspondingly. In addition, the optimal DBN manner has obtained moderate performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.993, 0.966, 0.982, 0.980, and 0.961 correspondingly. Besides, the optimal FFNN technique has resulted in competitive outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.996, 0.979, 0.989, 0.991, and 0.974 correspondingly. Finally, the MSSA-XGBoost technique has outperformed the other methods with the maximal sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.998, 0.985, 0.994, 0.996, and 0.981 correspondingly.

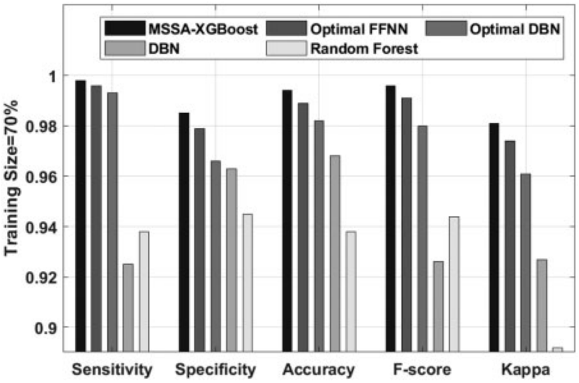

Figure 6: Comparative analysis of MSSA-XBoost technique under TS = 70%

A detailed results analysis of the MSSA-XBoost manner takes place under TS of 80% in Fig. 7. The outcomes exhibited that the RF system has accomplished worse performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.945, 0.953, 0.945, 0.953, and 0.912 correspondingly. Concurrently, the DBN manner has gained certainly higher outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.933, 0.978, 0.974, 0.929, and 0.931 respectively. Followed by, the BMO-DBN methodology has attained moderate performance with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.996, 0.972, 0.988, 0.984, and 0.966 respectively. Additionally, the BSA-FFNN system has resulted in competitive outcome with the sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.998, 0.988, 0.992, 0.996, and 0.978 respectively. However, the MSSA-XGBoost technique has demonstrated the other manners with the superior sen_y, spe_y, acc_y, F_{score}, and $kappa$ of 0.999, 0.992, 0.996, 0.997, and 0.989 correspondingly.

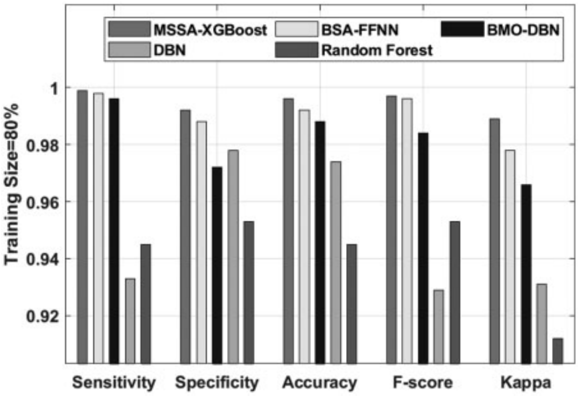

Figure 7: Comparative analysis of MSSA-XBoost technique under TS = 80%

The above mentioned results analysis demonstrated that the proposed manner is found to be an effective tool for data offloading and cyberattacks identification in MEC systems.

4 Conclusion

In this article, an effective AIMDO-SMEC technique has been developed to accomplish effective data offloading and security in MEC. The presented AIMDO-SMEC manner encompasses three major stages namely SNN based traffic flow prediction, ASCE based data offloading, and MSSA with XGBoost based cyberattack classification. The use of MSSA technique helps to properly adjust the parameters involved in the XGBoost technique and it results in improved detection outcomes. For examining the enhanced outcomes of the AIMDO-SMEC technique, a comprehensive experimental analysis is carried out and the results demonstrated the enhanced outcomes of the AIMDO-SMEC technique interms of different measures. The experimental results confirmed the supremacy of the AIMDO-SMEC technique over the recent approaches. In future, energy aware task scheduling and resource allocation models can be introduced for MEC systems.

Funding Statement: The authors extend their appreciation to the Deanship of Scientific Research at King Khalid University for funding this work under Grant Number (RGP 2/209/42). Princess Nourah bint Abdulrahman University Researchers Supporting Project Number (PNURSP2022R77), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia. The authors would like to acknowledge the support of Prince Sultan University for paying the Article Processing Charges (APC) of this publication.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] J. B. Wang, H. Yang, M. Cheng, J. Y. Wang, M. Lin *et al.,* "Joint optimization of offloading and resources allocation in secure mobile edge computing systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 8843–8854, 2020.

[2] Y. Zhou, P. L. Yeoh, C. Pan, K. Wang, M. Elkashlan *et al.,* "Offloading optimization for low-latency secure mobile edge computing systems," *IEEE Wireless Communications Letters*, vol. 9, no. 4, pp. 480–484, 2020.

[3] I. A. Elgendy, W. Zhang, Y. C. Tian and K. Li, "Resource allocation and computation offloading with data security for mobile edge computing," *Future Generation Computer Systems*, vol. 100, pp. 531–541, 2019.

[4] A. Rahman, E. Hassanain and M. S. Hossain, "Towards a secure mobile edge computing framework for hajj," *IEEE Access*, vol. 5, pp. 11768–11781, 2017.

[5] Y. Chen, Y. Zhang, S. Maharjan, M. Alam and T. Wu, "Deep learning for secure mobile edge computing in cyber-physical transportation systems," *IEEE Network*, vol. 33, no. 4, pp. 36–41, 2019.

[6] H. Yang, J. B. Wang, M. Cheng, C. Chang, J. Y. Wang *et al.,* "Secure resource allocation in mobile edge computing systems," in *2019 IEEE Global Communications Conf. (GLOBECOM)*, Waikoloa, HI, USA, pp. 1–6, 2019.

[7] T. Bai, J. Wang, Y. Ren and L. Hanzo, "Energy-efficient computation offloading for secure uav-edge-computing systems," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 6, pp. 6074–6087, 2019.

[8] X. Lai, L. Fan, X. Lei, Y. Deng, G. K. Karagiannidis *et al.,* "Secure mobile edge computing networks in the presence of multiple eavesdroppers," *IEEE Transactions on Communications*, vol. 70, pp. 1–1, 2021, https://doi.org/10.1109/TCOMM.2021.3119075.

[9] S. Lai, R. Zhao, S. Tang, J. Xia, F. Zhou *et al.,* "Intelligent secure mobile edge computing for beyond 5G wireless networks," *Physical Communication*, vol. 45, pp. 101283, 2021.

[10] A. M. Hilal, M. A. Alohali, F. N. A. Wesabi, N. Nemri, H. J. Alyamani *et al.,* "Enhancing quality of experience in mobile edge computing using deep learning based data offloading and cyberattack detection technique," *Cluster Computing*, 2021, https://doi.org/10.1007/s10586-021-03401-5.

[11] I. A. Elgendy, A. Muthanna, M. Hammoudeh, H. Shaiba, D. Unal *et al.,* "Advanced deep learning for resource allocation and security aware data offloading in industrial mobile edge computing," *Big Data*, vol. 9, no. 4, pp. 265–278, 2021.

[12] G. Mitsis, P. A. Apostolopoulos, E. E. Tsiropoulou and S. Papavassiliou, "Intelligent dynamic data offloading in a competitive mobile edge computing market," *Future Internet*, vol. 11, no. 5, pp. 118, 2019.

[13] M. S. Alkatheiri, "PCOS—Privacy-controlled offloading scheme for secure service data offloading in edge-internet of things-cloud scenario," *Arabian Journal for Science and Engineering*, 2021.

[14] Z. Xu, X. Liu, G. Jiang and B. Tang, "A Time-efficient data offloading method with privacy preservation for intelligent sensors in edge computing," *EURASIP Journal on Wireless Communications and Networking*, vol. 2019, no. 1, pp. 236, 2019.

[15] M. V. Arisoy, "Signature verification using siamese neural network one-shot learning," *International Journal of Engineering and Innovative Research*, 2021, https://doi.org/10.47933/ijeir.972796.

[16] T. Gopalakrishnan, D. Ruby, F. A. Turjman, D. Gupta, I. V. Pustokhina *et al.,* "Deep learning enabled data offloading with cyber attack detection model in mobile edge computing systems," *IEEE Access*, vol. 8, pp. 185938–185949, 2020.

[17] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. of the 22nd ACM SIGKDD Int. Conf. on Knowledge Discovery and Data Mining*, San Francisco California USA, pp. 785–794, 2016.

[18] I. U. Khan, N. Aslam, T. Anwar, S. S. Aljameel, M. Ullah *et al.,* "Remote diagnosis and triaging model for skin cancer using efficientnet and extreme gradient boosting," *Complexity*, vol. 2021, pp. 1–13, 2021.

[19] S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris *et al.,* "Salp swarm algorithm: A bio-inspired optimizer for engineering design problems," *Advances in Engineering Software*, vol. 114, pp. 163–191, 2017.

[20] N. Panda and S. K. Majhi, "Oppositional salp swarm algorithm with mutation operator for global optimization and application in training higher order neural networks," *Multimedia Tools and Application*, vol. 80, 2021, https://doi.org/10.1007/s11042-020-10304-x.