

Article

An Efficient and Secure Model Using Adaptive Optimal Deep Learning for Task Scheduling in Cloud Computing

Sahar Badri ¹, Daniyal M. Alghazzawi ¹, Syed Humaid Hasan ², Fayez Alfayez ³, Syed Hamid Hasan ^{1,*}, Monawar Rahman ⁴ and Surbhi Bhatia ^{5,*}

¹ Department of Information Systems, College of Computer Sciences and Information Technology, King Abdulaziz University, Jeddah 80200, Saudi Arabia; skbadri@kau.edu.sa (S.B.); dghazzawi@kau.edu.sa (D.M.A.)

² ACS Solution, Software Engineer—Loaned out to IBM Cloud, Minneapolis, MN 55437, USA; hasanqp@gmail.com

³ Department of Computer Science and Information, College of Science, Majmaah University, AL-Majmaah 11952, Saudi Arabia; f.alfayez@mu.edu.sa

⁴ Department of Information Technology, University of Technology and Applied Sciences, Al Musannah 516, Oman; monawar.rahman@yahoo.in

⁵ Department of Information Systems, College of Computer Sciences and Information Technology, King Faisal University, Al-Ahsa 31982, Saudi Arabia

* Correspondence: shhasan@kau.edu.sa (S.H.H.); sbhatia@kfu.edu.sa (S.B.)

Abstract: Recently, cloud computing resources have become one of the trending technologies that permit the user to manage diverse resources and a huge amount of data in the cloud. Task scheduling is considered one of the most significant challenges and ineffective management results in performance degradation. It is necessary to schedule the task effectively with maximum resource utilization and minimum execution time. Therefore, this paper proposes a novel technique for effective task scheduling with enhanced security in the cloud computing environment. A novel convolutional neural network optimized modified butterfly optimization (CNN-MBO) algorithm is proposed for scheduling the tasks, thereby maximizing the throughput and minimizing the makespan. Secondly, a modified RSA algorithm is employed to encrypt the data, thereby providing secure data transmission. Finally, our proposed approach is simulated under a cloudlet simulator and the evaluation results are analyzed to determine its performance. In addition to this, the proposed approach is compared with various other task scheduling-based approaches for various performance metrics, namely, resource utilization, response time, as well as energy consumption. The experimental results revealed that the proposed approach achieved minimum energy consumption of 180 kWh, a minimum response time of the 20 s, a minimum execution time of 0.43 s, and maximum utilization of 98% for task size 100.

Keywords: task scheduling; convolutional neural network; modified butterfly optimization; security



Citation: Badri, S.; Alghazzawi, D.M.; Hasan, S.H.; Alfayez, F.; Hasan, S.H.; Rahman, M.; Bhatia, S. An Efficient and Secure Model Using Adaptive Optimal Deep Learning for Task Scheduling in Cloud Computing. *Electronics* **2023**, *12*, 1441. <https://doi.org/10.3390/electronics12061441>

Academic Editors: Celestine Iwendu and Thippa Reddy Gadekallu

Received: 19 February 2023

Revised: 12 March 2023

Accepted: 14 March 2023

Published: 17 March 2023



Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Resource allocation is defined as the process of scheduling and assigning resources and the advantage of the combined form of resource allocation and task scheduling is to reduce the entire system delay. The virtualization platform is used for resource allocation and the resources such as output and input devices, central processing unit (CPU), and memory will be virtualized through the virtualization platform. Due to the virtualization, the system has some improvements such as minimum execution time, energy minimization, and effectiveness of cloud resources in the data centers [1]. The various issues such as energy management, server consolidation, data security, data management, managing automated services, and traffic problems mainly damage the performance, security, and availability of cloud computing [2]. The policies in resource allocation can identify the perfect task requirements before the utilization of the virtual machine. This policy will

manage the availability of resources, and eliminate service level agreement violations and task deadlines. The standard techniques are employed to overcome the problems in task scheduling in cloud computing [3].

The scheduling problems in cloud computing are mainly concentrated around the task allocation and virtual servers. Security and energy consumption are the key elements of task scheduling, storage of data and information, and the processing of modern cloud systems. Blockchain technology can act as a secure and effective technology and the data can be stored in a small database instead of using a central data center for data storage. This will increase security and prevents database damage from attacks on the entire cloud system [4]. The large-scale difficult applications have various service components that require distinctive combinations of reliability and security. The various stages of reliability and security schemes are required on the cloud side. Due to the inadequate security service, the multi-cloud system can eliminate the inefficient adaptation between cloud services and tasks, information leakage, and extreme level of computing time [5]. The task allocation with resource problems falls under the NP-hard problem category and the quality of service will solve task scheduling problems in cloud computing [6].

For the development of storing, sensing, and computing, a large amount of data is produced, which has difficulties in implementation and design of security-critical systems [7]. Cloud computing technology is eminent for the processing of big data, which contains computing problems and data storage problems [8]. Cloud computing contains performance and security issues in resource allocation, which requires improvements for increasing the quality and reducing the computational cost. The storage of data and processing of data are implemented by using private clouds through the public clouds [9]. Cloud computing includes various phases such as parallel computing, utility computing, distributed computing, and grid computing, which results in the rapid growth of data centers. Due to rapid growth, some issues occurred such as performance reduction and response delay. The combined form of cloud computing and green communication technology can be used for developing the data center's computing capabilities [10]. The process of scheduling and allocating resources and the advantage of an integrated form of resource allocation and task scheduling is to reduce overall system latency. Researchers developed several techniques for task scheduling algorithms. Meanwhile, these techniques have various issues such as energy management, server consolidation, data security, data management, managing automated services, and traffic problems, mainly damage the performance, security, and availability of cloud computing. This paper proposes a novel technique for effective task scheduling with enhanced security in the cloud computing environment.

This research paper accomplishes the following contributions.

- To design a task scheduling process using a novel convolutional neural network optimized modified butterfly optimization (CNN-MBO) algorithm thereby minimizing the response time and maximizing the resource utilization.
- Minimizing the total response time and makespan via optimal task scheduling process.
- A modified RSA algorithm is employed to encrypt the data, thereby providing secure data transmission in the cloud system.
- To compare the proposed approach with various other task scheduling-based approaches for various performance metrics, namely, resource utilization, response time as well as energy consumption.

The rest of this paper is organized as follows: Section 2 explains the literature survey of various authors. The proposed methodology is described in Section 3 and this section has a brief explanation of the CNN-MBO method. Section 4 contains the result and discussion part and the conclusion part is explained in Section 5.

2. Related Works

Rekha et al. [11] proposed an efficient task allocation method for a cloud environment employing a genetic algorithm. An effective task allocation method was employed for acquiring minimum completion time. For task scheduling, throughput and makespan were

the parameters utilized in this paper. The experimentation results demonstrated that the scheme achieved higher performance than the other methods. Energy consumption was not considered in this paper.

Sun et al. [12] discussed task scheduling security in the cloud environment. The functional domain construction was utilized for determining the sets of fog nodes included in the functional domain. The service time, success rate, average delay, and cost are the parameters employed in this paper. The results showed that the scheme achieved better performance when compared to the other techniques. However, this paper failed to focus on the reliability of service in the fog-clouds environment.

Nawrocki et al. [13] presented a security-aware task allocation model for resource management and secure task allocation in mobile cloud computing (MCC). To meet the consumers' data confidentiality requirements, they developed a resource allocation protocol that assigns optimal and secure communication. A machine learning model is adopted for resource utilization and their model is evaluated in a MocSecSim environment. The results show that their model enhances the security level for task allocation with minimal processing time and energy consumption.

Li, W. et al. [14] presented an agent learning method in a trust basis form in a mobile cloud computing atmosphere that involved service composition. This technique was developed to increase user skills such as risks in security and provision of quality of service (QoS) uncertainty. The efficiency was analyzed by applying Java Agent Development Framework (JADE). The experimentation result showed that the scheme achieved improvement in the satisfaction of users and success rate for transactions through the fuzzy c-means (FCM) learning algorithm.

The cloud computing and ant colony optimization algorithm (ACOA) was developed by Su, Y. et al. [15] to schedule tasks and allocate resource optimization. The cost and time expectation was achieved through optimized ACOA (Q-ACOA). The time, cost, and load balance-enhanced ant colony optimization (TCLB-EACO) algorithm, min-min (MM) algorithm, and the round robin scheduling (RR) algorithms were used in this scheme. The experimentation result demonstrated that the scheme evaluated the value of expected heuristic factor β is 4.5 and pheromone heuristic factor α is 3, which consumed low cost in Q-ACOA over related algorithms. However, the scheme does not execute the correlation process.

Amer, A.A. et al. [16] investigated an optimized collaborative scheduling (OCS) algorithm in a cloud computing and mobile-edge computing (MEC) scheme which involved resource-sharing methods for performing task prioritization. The presence of service satisfaction was verified by a resource optimization algorithm and the supply of resources was established. The experimentation results validated that the scheme achieved the best performance in a proposed scheduling algorithm over the previous scheduling algorithm. However, the scheme consumed more power and energy.

Zhu, Q.H. et al. [5] developed a multi-mound allocation (MMA) scheduling method called matching to optimize submitted tasks with respect to the total cost, makespan, reliability constraints, and security. Hybrid chaotic particle search (HCPS), modified artificial bee colony (MABC), and modified cuckoo search (MCS) are the techniques used to achieve these objectives. A better trade-off between time and economic cost, higher resource utilization lower cost, and shorter makespan is achieved using this technique.

The symmetric key cryptographic technique was proposed by Sohal, M. et al. [17] based on binary deoxyribonucleic acid (BDNA) and asymmetric deoxyribonucleic acid (ADNA) which involved data encryption on the client side for the analysis of cloud computing security. This technique was evaluated to reduce the usage of more encryption algorithms. The scheme related the developed technique with the existing Data Encryption Standard (DES), Advanced Encryption Standard (AES), Blowfish, and deoxyribonucleic acid (DNA) symmetric key algorithms. The experimentation analysis revealed that the scheme reached better encryption time, throughput, and ciphertext size than symmetric

key algorithms. Table 1 describes the literary works of task scheduling in cloud computing based on several authors.

Table 1. Literature survey based on several authors.

Author and Year	Technique Used	Pros	Cons
Rekha et al. (2019) [11]	Genetic Algorithm (GA)	Completion time was low	Energy consumption was not considered
Sun et al. (2021) [12]	Functional Domain construction	Low latency, and minimal service time	Does not focus on the reliability of service in the fog-clouds environment
Nawrocki et al. (2022) [14]	Machine Learning model (ML)	Enhances the security level for task allocation with minimal processing time	The service cost was high
Li, W. et al. (2019) [15]	Fuzzy C-Means (FCM) learning algorithm.	The security level was high	Low success rate
Su, Y. et al. (2021) [16]	Time, Cost, and Load Balance-Enhanced Ant Colony Optimization (ACO)	Low cost and Execution time	Does not execute the correlation process
Amer, A.A. et al. (2022) [17]	Optimized Collaborative Scheduling (OCS) algorithm	Provide high security	Consumed more energy and power
Zhu, Q.H. et al. (2021) [5]	Modified Artificial Bee Colony (MABC), and modified cuckoo search (MCS)	The better trade-off between time and economic cost	The accuracy level was low
Sohal, M. et al. (2022) [17]	Encryption Standard (DES) and Advanced Encryption Standard (AES)	Achieved higher throughput	Execution time was high

3. Proposed Methodology

The significant objective of this paper is to design an optimal task-scheduling process with minimum response time, energy consumption, as well as maximum resource utilization. The workflow of the proposed approach containing various task managers is depicted in Figure 1, in which the workload is split into numerous clusters. This paper proposes a CNN-optimized MBO approach for optimal scheduling of tasks and modified RSA is employed to secure the data stored in the cloud storage system. The following section depicts the detailed steps involved in the proposed approach.

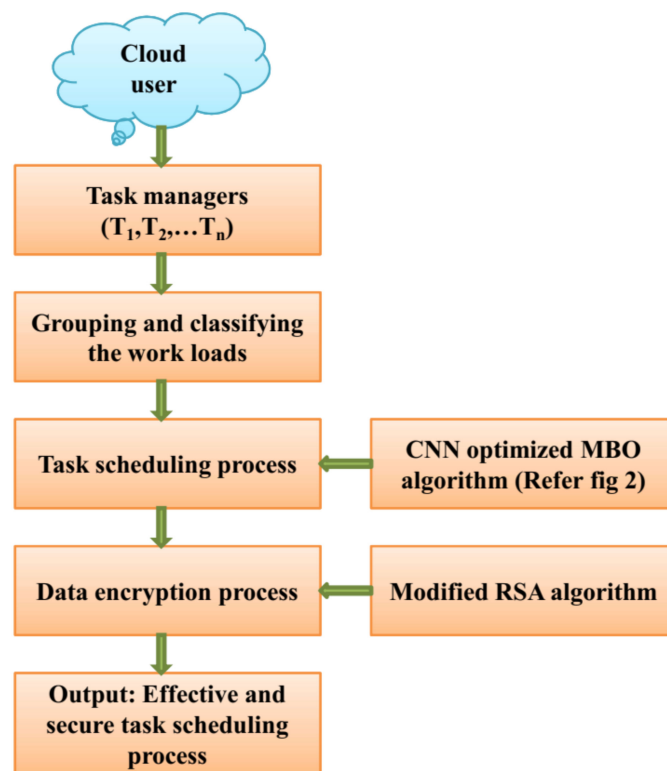


Figure 1. Proposed workflow.

3.1. Task Scheduling Process Based on CNN-MBO Algorithm

The task scheduling process is optimized using several parameters of multiple service providers such as user request, task type, task dependency, etc. Initially, the user request is set up, which consists of 1 to N task units. Then the task type is defined that is comprised of 1 to t tasks [18]. The term $t_{\max}$ indicates the total number of tasks that exist in the task unit. The task dependency t_{δ} achieved by the task unit is considered as t_{δ}^{xy} .

$$t_{\delta}^{xy} = \begin{bmatrix} & u_{R_1} & u_{R_2} & u_{R_3} \\ u_{R_1} & 0 & 1 & 0 \\ u_{R_2} & 1 & 0 & 0 \\ u_{R_3} & 0 & 1 & 0 \end{bmatrix} \quad (1)$$

The term u_{R_1} indicates dependencies of task units. Therefore, to enhance the performance of task scheduling in a cloud computing environment, we adopted the CNN-MBO algorithm. It not only enhances scheduling performance but also reduces time loss. A detailed description of the proposed CNN-MBO approach for task scheduling is deliberated as follows in Figure 1.

3.1.1. Convolutional Neural Network (CNN)

One of the types of deep neural networks (DNN) is utilized for classifying and analyzing images. Information extraction and processing vary by CNN. Generally, CNNs have convolutional layers, pooling layers, and fully connected layers. Convolutional layers have kernels that slide down and across the input to produce output by multiplying each weight in the kernel with the corresponding matrix element value over which the kernel acts. The result is then summed to obtain the output. The kernel has weights that are updated through backpropagation during the training process. Pooling layers are used to reduce the dimensionality of the feature maps and for more robust feature detection. CNNs have seen limited use in task scheduling problems compared to other neural networks [19].

Convolutional Layer

One of the CNN core layers has a convolutional layer. These layers are smaller in size and they have filters but they cover the entire image by shifting. The convolutional mechanism takes place by calculating the dot product among the filter and image. The dot products among the filter and image are summarized through the filter region.

$$a_k^m = \lambda \left(y_j^{m-1} \times x_{jk}^{(1)m} + c_k^{(1)m} \right) \quad (2)$$

Pooling Layer

The pooling layer operated downsampling. Different classes of the pooling function exist. The most commonly employed functions are those of maximum pooling. The maximal pooling filters returned the maximal values for every subregion. If the $2 \times 2 \times 1$ maximal pooling filters are utilized for the size feature $4 \times 4 \times 1$, then it resulted in a $2 \times 2 \times 1$ size feature.

Fully Connected Layer

The preceding layer neurons are associated with each neuron with subsequent layers in the fully connected layer. Then it is expressed as:

$$z_k^u = \lambda \left(a_j^{m-1} \times x_{jk}^{(2)u} + c_k^{(2)m} \right) \quad (3)$$

The bias is represented by $c_k^{(2)}$, and the input through the previous layer being denoted a_k^{m-1} . In the fully connected layer, the hidden layer sparsity will avoid the problem of overfitting in CNN.

Softmax Function Layer

The probability distributions of the event with various events are computed by using the softmax function layer. The softmax layer's operation is calculated and expressed in the following equation:

$$Q(z_k^m) = \frac{\exp(z_k^m)}{\sum_{k=1}^l \exp(z_k^m)} \quad (4)$$

Classification Output Layer

The loss is computed during the training of the output layer of the CNN. The goal of CNN is to reduce the cost function for predicting the data. Then the existent cost function is calculated and expressed in the following function:

$$e^{EX}(x, c) = c_E + \gamma \sum x^2 \quad (5)$$

Then the cross-entropy is expressed as:

$$c_E = - \sum_{k=1}^n z_k^u \ln z_k^q \quad (6)$$

The target value is represented as z^u and the predicted value is denoted z^q . The weight function of CNN is optimized using a modified butterfly optimization algorithm which is discussed as follows.

3.1.2. Modified Butterfly Optimization Algorithm

Most traditional optimization methods such as cuckoo search, firefly optimization, artificial bee colony, particle swarm, and genetic algorithm are utilized to find the optimal solution. However, these techniques rely on slope information to update the solution, and the position of the initial solution affects the final solution quality. Among all methods,

the BOA algorithm provides better performance. The BOA algorithm is a nature-inspired algorithm and has been widely used to solve complex optimization problems in various fields. The butterfly optimization algorithm (BOA) revealed its efficiency with the robust and straightforward strategy to acquire optimum solutions to different optimization difficulties—it has particular deficiencies. Due to the switching probability of higher values, greater numbers of candidates are utilized for the global search. In this algorithm, some of the candidates are moving to the local search stage, which constructs the algorithm weakest in local searchability. This algorithm has suffered from premature convergence and it does not avoid the local optima. The Steven power law is utilized for updating the exponent of power value. For enhancing the exploration capability, Lagrange interpolation is utilized and the local searching scheme is updated by the strategy of Lévy flight. The modified BOA (MBOA) was employed for balancing local exploitation and global exploration [18]. Then the modifications are described below:

The BOA switching probability is customized for increasing the efficiency of this algorithm. An arbitrary selection of exploitation and exploration phases based on the chosen value of switching probability is recognized to sometimes cause the BOA to misplace direction and deviate from the global optimal solution. Thus, this paper proposed a convolutional neural network optimized modified butterfly optimization (CNN-MBO) algorithm to enhance the exploitation and exploration capacity effectively and better balance the algorithm in two stages of switching [20].

Primarily, the value q is taken as 0.3. Due to the scattered position or randomness of butterflies, more of the butterflies will not sense the best butterfly's fragrance. The local stage of the BOA is termed by the randomly moving butterflies. The switch probabilities are expressed and calculated in the following equation:

$$q = 0.4 + 0.3 \times \left(\frac{C_I}{T_I} \right) \quad (7)$$

Modification of Fragrance Generation

Every butterfly fragrance generation is modified by the exponent power values as well as a sensory modality. The exponent power is taken in the ranges of [0.2, 0.6]. Based on Steven's power law, the power exponent value for fragrance is fixed as 0.6. The smaller number of butterflies received the fragrance with the best butterflies and the total number of butterflies is enhanced by receiving the fragrance with the best butterfly. The new changed fragrance is calculated and generated utilizing the following equation:

$$g(j) = d \times \left(\frac{h_{BEST} - J_j}{C_I \times J_{MEAN}} \right)^b \quad (8)$$

The sensor modality is represented by d , the objective of the best solution is indicated by h_{BEST} , the current iteration number is denoted by C_I , and the mean objective values and current objective functions are denoted by J_{MEAN} and J_j . The exponent power and the sensor modality are represented by b and d .

$$b = 0.2 + (0.6 - 0.2) \times \frac{I_C}{T_I} \quad (9)$$

$$d = 0.01 + 0.025 \times \frac{I_C}{T_I} \quad (10)$$

Lagrange Interpolation

The Lagrange interpolation is utilized for reducing the exploitation ability of the algorithm. The three-point Lagrange interpolation is employed for increasing the ability of the algorithm. The Lagrange interpolation application with the global stage of the BOA ensured the ability of exploitation in the stage.

Lévy Flight Searching Process

In the mLBOA, the exploitation stage is modified and expressed as below;

$$C_G_j^{u+1} = C_G_j^u + (s \times \text{SGN}(s - 0.6)) \times \text{LEVY}(1, E, 0.5) \quad (11)$$

The dimension is represented by E , the Lévy flight search method or Lévy flight random walk is multiplied through the random number ($s \times \text{SGN}(s - 0.5)$). The search orientation is evaluated by the direction vector value l .

$$q = 0.4 + 0.3 \times \left(\frac{C_l}{T_l} \right) \quad (12)$$

$$Q(y_k) = B(y_j)^2 + C(y_j) + D \quad (13)$$

$$C_G_j^{u+1} = C_G_j^u + (s \times \text{SGN}(s - 0.5) \times \text{LEVY}(1, E, 0.5)) \quad (14)$$

3.1.3. Proposed CNN-Optimized MBO Approach for Efficient Task Scheduling

Figure 2 illustrates the flow diagram of the proposed CNN-optimized MBO approach for an efficient task scheduling process. At first, the input dataset is fed into the neural network that extracts the significant data features. In CNN, the inputted data are flattened to vectors before passing to a fully connected layer. Generally, the highly trained fully connected layer necessitates many parameters, which create over-fitting problems. Therefore, to provide continuous representation and optimize weight function, the modified butterfly optimization (MBO) algorithm is employed. Thus, the CNN-MBO algorithm optimizes the task-scheduling process in an effective manner.

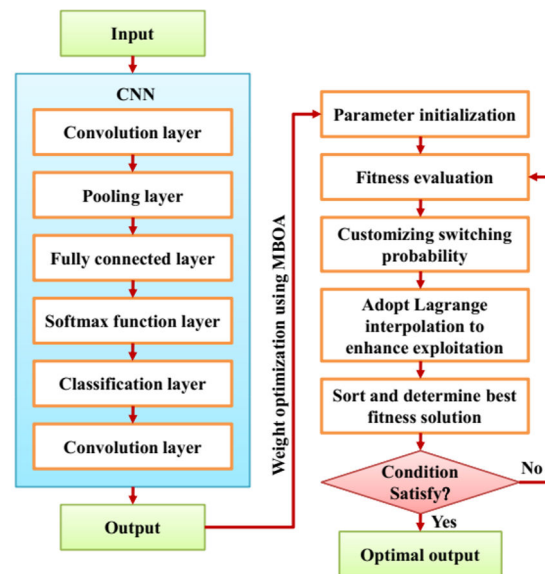


Figure 2. Flow diagram for CNN-MBO approach for efficient task scheduling.

3.1.4. Complexity Analysis

This section focus on studying the time complexity using big-O notation, therefore the approach with higher speed will be presented. The population size, N , the number of tasks, n , and the maximum iteration are the main elements that have higher impact on speed. The time complexity of HDE in big-O is $O(T * n * N)$.

3.2. Modified RSA (MRSA) Based Encryption Process

Information leakage (data loss) is the most significant cause of security issue that arises in cloud computing task. In order to tackle this security issue, the data encryption approach

is necessary. The data encryption approach safeguards sensitive information and private data from unauthorized users by providing safe and secure cipher keys. In this work, data are securely stored in the cloud and communicated between the user and the server by adopting a modified RSA. A brief illustration of this technique is described as follows.

Modified RSA

In this paper, a modified RSA (MRSA) algorithm is employed for storing secure data in the cloud storage system. The MRSA algorithm comprises three vital phases, namely, the key generation phase, the encryption phase, as well as decryption phase [21]. For the RSA encryption and decryption based on BOA, the RSA algorithm requires modification of BOA components to improve speed. The modified BOA algorithm is used in MRSA to reduce the processing time of encryption and decryption. The MRSA algorithm provides more security and speed in the encryption and decryption process as compared to RSA.

A detailed description of each respective phase involved in the MRSA algorithm is discussed below.

Key Generation Process

In an asymmetric key cryptosystem, a set of private and public keys must be created by every user. For this, decryption uses one key and encryption uses the other key.

1. a, b, c , and d are the four distinct prime numbers chosen, i.e., $a \neq b \neq c \neq d$.
2. Evaluate $R = a \times b$ and $S = c \times d$.
3. Evaluate $\Phi(R) = (a - 1) \times (b - 1)$ and $\Phi(S) = (c - 1) \times (d - 1)$ and then create the primary numbers.
4. Select 2 public keys, $T2$ from $\Phi(S)$ and $T1$ from $\Phi(R)$. The condition below must be fulfilled when finding those values. $1 < T1 < \Phi(R)$ and $\text{QKL}(\Phi(R), T1) = 1$. This is related to another $T2$ value.
5. The extended Euclid's algorithm is used to establish $L1$ and $L2$ private keys.

Encryption Process

During the encryption process, the plain text or secret data are encrypted using two diverse public keys, P_1 and P_2 , by employing the modular exponential technique. The formula to encrypt the data is mentioned in equation.

$$\zeta = \left[\rho^{P_1} \mod n \right]^{P_2} \mod m \quad (15)$$

From the above equation, ζ and ρ signify the chipper text and plain text, respectively.

Decryption Process

The secret data received are unscrambled by employing the modular exponential technique as mentioned above by employing two private keys, namely, M_2 . The notation and description of variables are presented in Table 2.

$$\rho = \left[\zeta^{P_1} \mod m \right]^{M_2} \mod n \quad (16)$$

Table 2. Notations and description of variables.

Notations	Description
t_δ	Task dependency
u_{R_1}	Dependency of a task unit
$c_k^{(2)}$	Bias
a_k^{m-1}	The input of the previous layer
z^U	Target value

Table 2. *Cont.*

Notations	Description
z^q	Predicted value
d	Sensor modality
h_{BEST}	The objective of the best solution
C_I	Current iteration number
J_{MEAN}	Mean objective value
J_j	Current objective function
b	Exponent power
E	Dimension
ζ	Chipper text
ρ	Plain text

4. Results and Discussion

This section describes the analysis to determine the efficiency of the proposed convolutional neural network-based modified butterfly optimization (CNN-MBO). Metrics such as resource utilization, response time, energy consumption, and execution time are computed. The proposed CNN-MBO is simulated in a cloudlet simulator.

4.1. Experimental Setup

The experiments of the proposed CNN-MBO method are evaluated with various components and their specifications are depicted in Table 3.

Table 3. Experimental configuration.

Component	Specifications
Operating system	Windows (X86 ultimate) 64-bit OS
Processor	Intel Pentium CPU G2030 @ 3.00 GHZ
System	64 Bit OS system
RAM	4 GB
Hard disk	1 TB

4.2. Parameter Settings

The parameters utilized to tune the proposed method are delineated in Table 4.

Table 4. Parameter settings.

Techniques	Parameters	Ranges
CNN	Learning rate	0.005
	Number of hidden layers	7
	Number of epochs	3
	Activation function	Sigmoid
	Batch size	32
	Dropout	0.5
MBO	Switching probability	0.3 to 0.7
	Total number of iterations	100
	Sensor modality	0.01 to 0.03
	Exponent power	0.2 to 0.6

4.3. Performance Metrics

4.3.1. Resource Utilization

Resource utilization is defined as the total number of allocated resources used by a task to finish its execution. The mathematical expression Re_U is given as:

$$Re_U = R_A - R_N \quad (17)$$

From Equation (17), the unutilized resource and available resource are denoted as R_N and R_A .

4.3.2. Energy Consumption

The amount of energy utilized while allocating resources in the cloud server is termed energy consumption.

4.3.3. Response Time

The elapsed time between the arrival time of task R_A and the completion time of task R_C is defined as the response time.

$$R_T = R_C - R_A \quad (18)$$

4.4. Performance Analysis

Figure 3 depicts the response time analysis with different task sizes. The proposed CNN-MBO model is compared with the fuzzy c-means learning (FCM) algorithm [22], the time, cost, and load balance-enhanced ant colony optimization (TCLB-EACO) algorithm, genetic algorithm (GA) [23], and optimized collaborative scheduling (OCS). The response time for a different number of task sizes is graphically represented. The proposed CNN-MBO model achieves a faster response time compared to other methods. For a low number of task sizes, the response time of the proposed CNN-MBO model is 20 s.

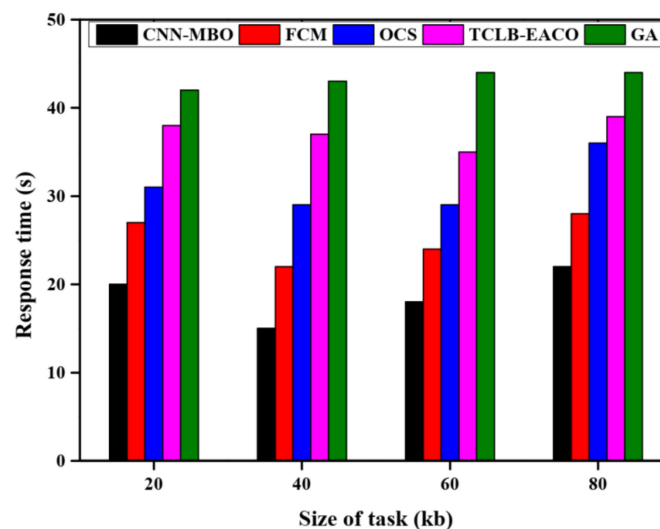


Figure 3. Response time analysis.

A comparative analysis of the execution time of various methods is represented in Figure 4. The proposed CNN-MBO model obtains minimum execution time, which provides effective results. The method has a higher execution time compared to other methods. The execution time reached by the proposed CNN-MBO model is about 0.43 s.

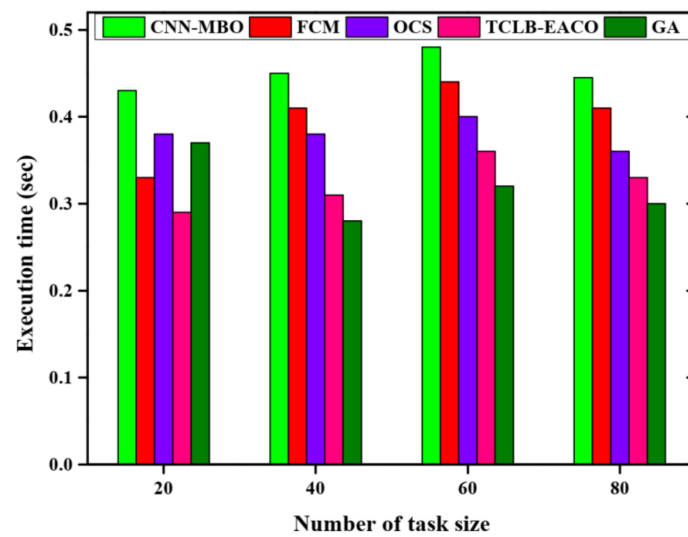


Figure 4. Execution time analysis.

Figure 5 represents the resource utilization evaluation of various methods. The resource utilization rate of the proposed CNN-MBO model is compared with the resource utilization rate of FCM, TCLB-EACO, GA, and OCS. The resulting analysis shows that the proposed CNN-MBO model has a higher resource utilization rate compared to existing methods. As the number of task sizes increases, resource utilization gradually increases. The resource utilization rate of the proposed CNN-MBO model is about 98% for 100 task sizes.

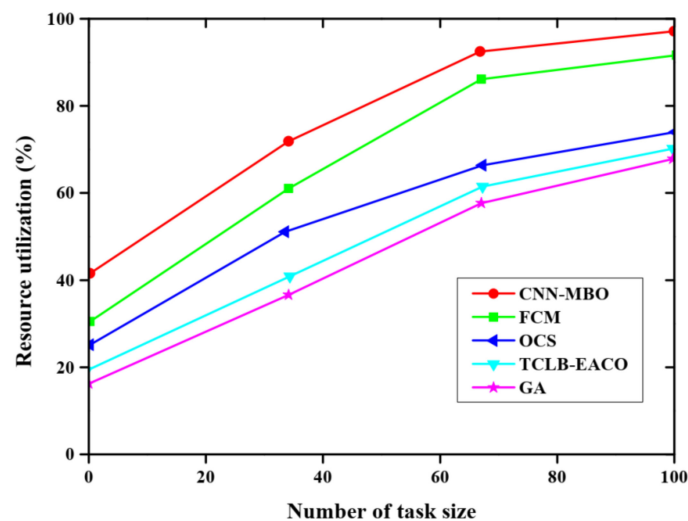


Figure 5. Resource utilization analysis.

Energy consumption analysis of various methods is given in Figure 6. The proposed method has achieved lower energy consumption compared to FCM, TCLB-EACO, GA, and OCS. Energy consumption for different task sizes is determined. When the task size is 20, the total energy consumed is 180 kWh.

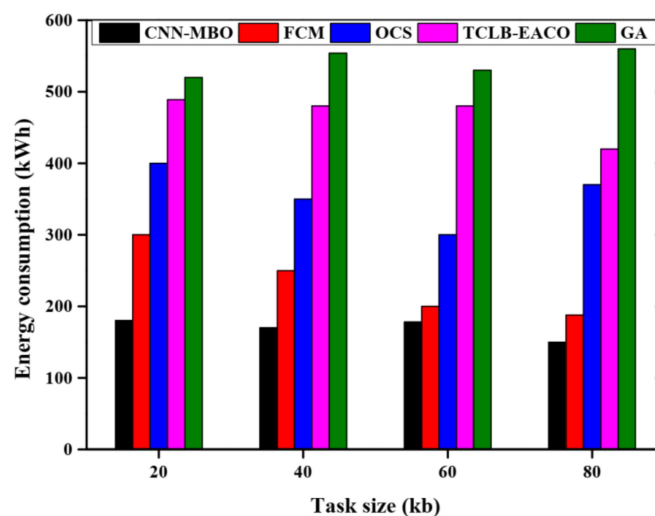


Figure 6. Energy consumption analysis.

5. Conclusions

This paper proposed a novel convolutional neural network optimized modified butterfly optimization (CNN-MBO) algorithm by minimizing the response time and maximizing the resource utilization. Here, a modified RSA algorithm is employed to encrypt the data, thereby providing secure data transmission. The proposed approach is simulated under a cloudlet simulator and the evaluation results are analyzed to determine its performance. Metrics such as resource utilization, response time, energy consumption, and execution time are computed to determine the effectiveness of the proposed CNN-MBO model. In the proposed work, the energy consumed is very low, and response time is also minimum. The resource utilization rate of the proposed CNN-MBO model is about 98% for 100 task size, which is higher than FCM, TCLB-EACO, GA, and OCS, respectively. However, the arrival time of the task is delayed. In the future, the proposed approach will be implemented in a real-time environment to determine the effectiveness of the system also to enhance the QoS based on task scheduling algorithms by considering the arrival time of the task.

Author Contributions: Conceptualization, S.B. (Sahar Badri) and M.R.; Methodology, D.M.A., S.H.H. (Syed Humaid Hasan), S.H.H. (Syed Hamid Hasan) and S.B. (Surbhi Bhatia); Validation, D.M.A. and M.R.; Formal analysis, S.B. (Sahar Badri) and D.M.A.; Investigation, S.H.H. (Syed Humaid Hasan) and S.H.H. (Syed Hamid Hasan); Resources, F.A. and S.B. (Surbhi Bhatia); Data curation, F.A.; Writing—original draft, D.M.A., S.H.H. (Syed Humaid Hasan), S.H.H. (Syed Hamid Hasan) and M.R.; Writing—review & editing, F.A. and S.B. (Surbhi Bhatia); Project administration, S.B. (Sahar Badri). All authors have read and agreed to the published version of the manuscript.

Funding: The Deanship of Scientific Research (DSR) at King Abdulaziz University (KAU), Jeddah, Saudi Arabia, has funded this project, under grant No. (RG-91-611-42).

Data Availability Statement: The data is available on request.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Praveenchandar, J.; Tamilarasi, A. Dynamic resource allocation with optimized task scheduling and improved power management in cloud computing. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 4147–4159. [\[CrossRef\]](#)
2. Walia, N.K.; Kaur, N. Performance Analysis of the Task Scheduling Algorithms in the Cloud Computing Environments. In Proceedings of the 2021 2nd International Conference on Intelligent Engineering and Management (ICIEM) 2021, London, UK, 28–30 April 2021; pp. 108–113.
3. Bharany, S.; Sharma, S.; Bhatia, S.; Rahmani, M.K.I.; Shuaib, M.; Lashari, S.A. Energy Efficient Clustering Protocol for FANETS Using Moth Flame Optimization. *Sustainability* **2022**, *14*, 6159. [\[CrossRef\]](#)
4. Abouyoussef, M.; Bhatia, S.; Chaudhary, P.; Sharma, S.; Ismail, M. Blockchain-Enabled online diagnostic platform of suspected patients of COVID-19 like pandemics. *IEEE Internet Things Mag.* **2021**, *4*, 94–99. [\[CrossRef\]](#)

5. Zhu, Q.H.; Tang, H.; Huang, J.J.; Hou, Y. Task scheduling for multi-cloud computing subject to security and reliability constraints. *IEEE/CAA J. Autom. Sin.* **2021**, *8*, 848–865. [\[CrossRef\]](#)
6. Shahzad, F.; Javed, A.R.; Zikria, Y.B.; Rehman, S.; Jalil, Z. Future smart cities: Requirements, emerging technologies, applications, challenges, and future aspects. *TechRxiv* **2021**. [\[CrossRef\]](#)
7. Zhou, J.; Sun, J.; Cong, P.; Liu, Z.; Zhou, X.; Wei, T.; Hu, S. Security-critical energy-aware task scheduling for heterogeneous real-time MPSoCs in IoT. *IEEE Trans. Serv. Comput.* **2019**, *13*, 745–758. [\[CrossRef\]](#)
8. Kummar, S.; Bhushan, B.; Bhatia, S. Blockchain Based Big Data Solutions for Internet of Things (IoT) and Smart Cities. In *New Trends and Applications in Internet of Things (IoT) and Big Data Analytics*; Springer: Cham, Switzerland, 2022; pp. 225–253.
9. Alghazzawi, D.; Bamasaq, O.; Bhatia, S.; Kumar, A.; Dadheech, P.; Albeshri, A. Congestion Control in Cognitive IoT-Based WSN Network for Smart Agriculture. *IEEE Access* **2021**, *9*, 151401–151420. [\[CrossRef\]](#)
10. Lu, Y.; Sun, N. An effective task scheduling algorithm based on dynamic energy management and efficient resource utilization in green cloud computing environment. *Clust. Comput.* **2019**, *22*, 513–520. [\[CrossRef\]](#)
11. Rekha, P.M.; Dakshayini, M. Efficient task allocation approach using genetic algorithm for cloud environment. *Clust. Comput.* **2019**, *22*, 1241–1251. [\[CrossRef\]](#)
12. Sun, Y.; Jin, R.; Lin, K.; Liu, W. High-performance isolation computing technology for smart IoT healthcare in cloud environments. *IEEE Internet of Things* **2021**, *8*, 16872–16879.
13. Nawrocki, P.; Pajor, J.; Sniezynski, B.; Kolodziej, J. Modeling adaptive security-aware task allocation in mobile cloud computing. *Simul. Model. Pract. Theory* **2022**, *116*, 102491. [\[CrossRef\]](#)
14. Li, W.; Cao, J.; Hu, K.; Xu, J.; Buyya, R. A trust-based agent learning model for service composition in mobile cloud computing environments. *IEEE Access* **2019**, *7*, 34207–34226. [\[CrossRef\]](#)
15. Su, Y.; Bai, Z.; Xie, D. The optimizing resource allocation and task scheduling based on cloud computing and Ant Colony Optimization Algorithm. *J. Ambient. Intell. Humaniz. Comput.* **2021**, 1–9. [\[CrossRef\]](#)
16. Amer, A.A.; Talkhan, I.E.; Ahmed, R.; Ismail, T. An Optimized Collaborative Scheduling Algorithm for Prioritized Tasks with Shared Resources in Mobile-Edge and Cloud Computing Systems. *Mob. Netw. Appl.* **2022**, *27*, 1444–1460. [\[CrossRef\]](#)
17. Sohal, M.; Sharma, S. BDNA-A DNA inspired symmetric key cryptographic technique to secure cloud computing. *J. King Saud Univ.-Comput. Inf. Sci.* **2022**, *34*, 1417–1425. [\[CrossRef\]](#)
18. Kumar, A.; Gandhi, C.P.; Zhou, Y.; Kumar, R.; Xiang, J. Improved deep convolution neural network (CNN) for the identification of defects in the centrifugal pump using acoustic images. *Appl. Acoust.* **2020**, *167*, 107399. [\[CrossRef\]](#)
19. Bal, P.K.; Mohapatra, S.K.; Das, T.K.; Srinivasan, K.; Hu, Y.C. A Joint Resource Allocation, Security with Efficient Task Scheduling in Cloud Computing Using Hybrid Machine Learning Techniques. *Sensors* **2022**, *22*, 1242. [\[CrossRef\]](#) [\[PubMed\]](#)
20. Sharma, S.; Chakraborty, S.; Saha, A.K.; Nama, S.; Sahoo, S.K. mLBOA: A Modified Butterfly Optimization Algorithm with Lagrange Interpolation for Global Optimization. *J. Bionic Eng.* **2022**, *19*, 1161–1176. [\[CrossRef\]](#)
21. Kalaichelvi, V.; Meenakshi, P.; Vimala Devi, P.; Manikandan, H.; Venkateswari, P.; Swaminathan, S. A stable image steganography: A novel approach based on modified RSA algorithm and 2–4 least significant bit (LSB) technique. *J. Ambient. Intell. Humaniz. Comput.* **2021**, *12*, 7235–7243. [\[CrossRef\]](#)
22. Bhatia, S. New improved technique for initial cluster centers of K means clustering using Genetic Algorithm. In *Proceedings of the International Conference for Convergence for Technology-2014*, Pune, India, 6–8 April 2014; pp. 1–4.
23. Bhalla, K.; Koundal, D.; Bhatia, S.; Rahmani, M.K.I.; Tahir, M. Fusion of infrared and visible images using fuzzy based siamese convolutional network. *Computers. Mater. Contin.* **2022**, *70*, 5503–5518.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.