*Intelligent Automation & Soft Computing*
DOI:10.32604/iasc.2022.022499
***Article***

Tech Science Press

# Hybrid Deep Learning Framework for Privacy Preservation in Geo-Distributed Data Centre

**S. Nithyanantham[1,*] and G. Singaravel[2]**

[1]Department of Computer Science and Engineering, Bannari Amman Institute of Technology, Sathyamangalam, 638401, India
[2]Department of Information Technology, K.S.R. College of Engineering (Autonomous), Tiruchengode, 637215, India
*Corresponding Author: S. Nithyanantham. Email: s.s.nithyanantham@gmail.com

Received: 10 August 2021; Accepted: 18 September 2021

**Abstract:** In recent times, a huge amount of data is being created from different sources and the size of the data generated on the Internet has already surpassed two Exabytes. Big Data processing and analysis can be employed in many disciplines which can aid the decision-making process with privacy preservation of users' private data. To store large quantity of data, Geo-Distributed Data Centres (GDDC) are developed. In recent times, several applications comprising data analytics and machine learning have been designed for GDDC. In this view, this paper presents a hybrid deep learning framework for privacy preservation in distributed DCs. The proposed model uses Deep Neural Network (DNN) for the feature extractor and classifier operations. In addition, Siamese training method is applied to fine-tune the prevention of secondary inference on the data. Moreover, gradient descent approach is employed to reduce the loss function of the DNN model. Furthermore, Glow-worm Swarm Optimization (GSO) algorithm is utilized to fine tune the hyperparameters of the DNN model to improve the overall efficiency. The proposed model is executed on a Hadoop based environment, i.e., Hadoop Distributed File System (HDFS), which has two nodes namely master node and slave nodes. The master node is considered as the main user node to get the services from the service provider. Every slave node behaves as per master node's instruction for data storage. In order to validate the enhanced performance of the proposed model, a series of simulations take place and the experimental results demonstrate the promising performance of the proposed model. The simple technique has reached a maximum gender recognition accuracy of 95, 90 and 79 on the applied data 1, 2 and 3 respectively. Also, the reduced simple approach has attained reduced gender recognition with the accuracy of 91, 84 and 74 on the applied data 1, 2 and 3 respectively.

**Keywords:** Data centre; big data analytics; deep learning; privacy preservation; gso algorithm; hadoop environment

## 1 Introduction

At present, with big data and further data generation, it is increasingly important to process and store largescale data in real-time that has led to the placement of cloud computing [1]. The rapid development of the Data Centre (DC) markets leads to significant development of energy utilization and cost [2]. The widespread performance of the cloud has led to pervasive transmission of cloud user, globally. Additionally, consideration of security, disaster recovery encourages organizations and globalization to share their DCs through distinct areas, near to cloud users and over a longer geographical distance [3]. This GDDC that interchanged the centralized one, offers solution to handle the larger volume and velocity of big data produced from geographically distributed sources [4].

Facebook has constructed 4 GDDC to manage and maintain these data. By raising generated data volumes in GDDC to allocate computations for using data locality turns into a developing area of research [5], instead of collecting each data needed for computation of an individual datacentre. As well, several data owners forbid transferring (raw) data out of datacentres for many reasons. Although, the cost of transferring data through geographical areas is generally higher because of the large volumes of data and the scarcity of cross datacentre's network bandwidth [6]. Storage devices, network infrastructure, and novel ways of addressing rising processing demands are all fueled by Big Data. Storage is one of the most critical infrastructural components of Big Data analytics. Conversely, laws and regulations like data privacy do not permit raw data to be transferred geographically [7]. Therefore, one significant challenge is how to efficiently and effectively "transfer computations to data" for improving the performances of data analysis and better use of the network bandwidth resource. Fig. 1 depicts the privacy preservation in big data environment.

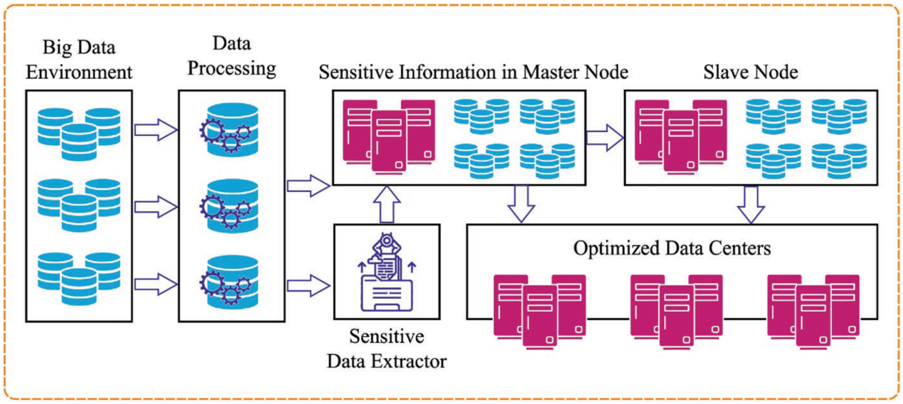

**Figure 1:** Privacy preservation in big data environment

At present, almost all the cloud provider offers GDDC cloud service for users to operate their application in many geographic positions [8]. E.g., Amazon EC2 presently has eleven service areas geographically dispersed across the globe, Google Compute Engine (GCE) and Windows Azure operate in 20 regions provide service from 4 GDDC. In recent times, several applications comprising machine learning [9] high performance computing and data analytics [10] have emerged for GDDC. Each application adopts a "transfer computations to data" model for using data locality and fulfils data privacy. Amongst geo-distributed applications, the "transfer computations to data" challenge for distributed and parallel data analysis program that corresponds to geo-distributed procedure mapping problems [11]. Several methods were introduced for optimizing the procedure mapping problems of random network and process topologies. In the geo-distributed cloud platform, data movement could be limited for distinct purposes like data privacy regulation [12]. Cloud based machine learning algorithms could offer useful services (for example health apps/video editing tools) [13].

This paper presents a hybrid Deep Learning (DL) framework for preserving the privacy in distributed DCs. The proposed model uses Deep Neural Network (DNN) for the feature extractor and classifier operations. To examine the better outcomes of the proposed model, a comprehensive experimental analysis is performed and the results are inspected in terms of different measures.

## 2 Related Work

Jeon et al. [14] proposed a distributed DL architecture for preserving the privacy in medicinal data training. To prevent patients' data leakage in medical platform, the hidden layer in the DL architecture is detached and the initial layers are saved in platforms and other layers are saved in a centralized server. While saving the patients' data in local platform, it maintains their privacy using the server. The succeeding layer enhances the learning efficiency through each data from every platform at the time of training. Kabir et al. [15] proposed a new method for achieving privacy preservation through SVM approach in which the training sets are dispersed and all partitions may have largescale data. The conventional SVM method is used for training the datasets of local data centres and the centralized machine calculates the concluding results. The presented method secures in an adverse platform and utilize simulation for demonstrating its accuracy and computational speed than other equivalent SVM training methods.

Taheri et al. [16] analysed the relation between GDDC power patterns using their weather parameter (according to distinct DC infrastructure and situations) and extracts a group of significant features. Next, the attained features are employed for providing an energy utilization predicting method which forecasts the power patterns of every cooperating DCs in a cloud. Tang et al. [17] developed a hybrid DRL based architecture that de-couples the VNF positioning and flow routing to distinct models. In the presented architecture, only DRL agents are accountable to learn the policies of VNF placement. Virtual Network Functions (VNFs), makes it possible for network operators to incorporate and reconfigure network capabilities quickly and easily. They customize the framework of the agent based on DDPG and adapt numerous methods for improving the learning efficacy, for example, wolpertinger policy, prioritized experience replays and adaptive parameter noise. The flow routing is performed in a GBM model. Also, they designed a decentralized routing approach for a GBM to address scalability.

Kaissis et al. [18] presented PriMIA, a free public domain software architecture for inference of medical imaging where the data is securely encrypted and differentially private. They tested PriMIA with a real-time analysis where an expert's level DCNN classifies paediatric chest Xrays, the resultant model classification performances are at the same level with non-securely, locally trained methods. Sajid et al. [19] examined the power requirement cost optimization problems in cloud datacentres and proposed a blockchain based decentralized task management and distribution methods. Furthermore, they minimized the request scheduling time to transfer each task from single datacentre to other and protect the power cost optimization procedure because of communication failure or shut down. The presented study is primitive for presenting the power cost optimization problems in GDDC by presenting blockchain based secured task scheduling approach considering the spatial and temporal variations related to task arrival procedure and electricity tariffs.

In Nithyanantham et al. [20], an MM-MGSMO approach is proposed. Now, using large data volumes and search space as input for GDDC, glow worms (viz., virtual machine) population are initialized. For all glow worms possessing a specific number of luciferins (viz., objective function), multiple objective functions (viz., storage capacity, bandwidth, computation and energy costs) are determined for all VMs. Then, the glow worm's location is upgraded based on the neighbouring factor through likelihood. Next, MapReduce functions identify the optimum VM and consequently, allocations are implemented, thus enhancing the data allocation efficacy. Also, the tasks are allocated through the datacentres, where the decrease in storage capacity and computation costs is assured. Zhou et al. [21] presented the special privacy requirement in GDDC and developed the geo-distributed procedure mapping problem as optimization problem using many limitations. Also, they developed a novel approach for effectively detecting a procedure mapping solution to the problems. The simulation result shows that the real cloud (includes Windows Azure and Amazon EC2) and simulation demonstrates the presented method could attain a considerable efficiency than advanced methods.

## 3 The Proposed Method

The proposed hybrid framework achieves privacy and optimization of resource allocation in big data services. Hybrid architecture collaborates master node and virtual machine by executing previously fine-tuned data in the big data framework. The proposed architecture is designed based on the idea that the user data computed is the sensitive information which can be misused by hackers and hence the master node is highly secured. The detailed working of the modules involved in the proposed model is discussed below.

### *3.1 Hadoop Environment*

With the continuous development of data, standard data investigation schemes could not save and produce a huge volume of data. Therefore, an optimum solution for managing the abundant data for storing it from the Hadoop distributed File System (HDFS) is employed. According to their fault tolerance process, the HDFS permits Hadoop to operate more reliable and efficiently. The HDFS is regarded as a regular file scheme the only variance is that it handles superior datasets. This technique separates data as 64 MB blocks by default, creating it additional effective to huge datasets. The data in HDFS are saved by two procedures namely the actual data and their metadata namely file location and file size. The master node has three elements namely the job tracker, name node and secondary name node but the slave node contains task tracker and data nodes.

### *3.2 DNN Model for Privacy Preservation*

The user likes to interact with big data service and a classifier method is utilized to have an essential primary knowledge of the user data. The sensitive data in the Virtual Machines (VM) depends on the classifier technique and the VM users provide security for big data services to preserve the sensitive data. The solution for this issue is to provide sensitive extractors on the master nodes and raw data is transmitted to VM. The removed raw data is regarded as an exclusive data that is considered to be an important task. The trained DNN has two parts placed in the service provider's model. If the HDFS assigns the master nodes, the sensitive extractor of DNN is sent to the master nodes for removal.

A person's right to decide who has access to their personal data and how that data is used is called privacy. The data is owned by the data holder, and as such, the data itself represents a danger to individual privacy. Social networking applications, websites, mobile apps, ecommerce sites, banks, and hospitals are just a few examples of data holders. The data holder bears the obligation of protecting the privacy of the users.

The structure of DNN basically has three primary modules that contain input, output and hidden layers. By assuming the efforts of preference weight fitness, the DNN is intended with two hidden layers for perfect learning of the mapping connection between input and output data. During the training stage, by using the JOA, the DNN iteratively upgrades the weight of nodes from the hidden layer. At the hidden layer, the total number of nodes are estimated by Eq. (1).

$$n = \sqrt{a + b} + c \tag{1}$$

where the number of input layer nodes are represented as ‘a’ the number of output layer nodes are represented as ‘b’, the number of hidden layer nodes are signified as n and *a* constant value between [1–10] is written as c. Fig. 2 illustrates the framework of DNN.

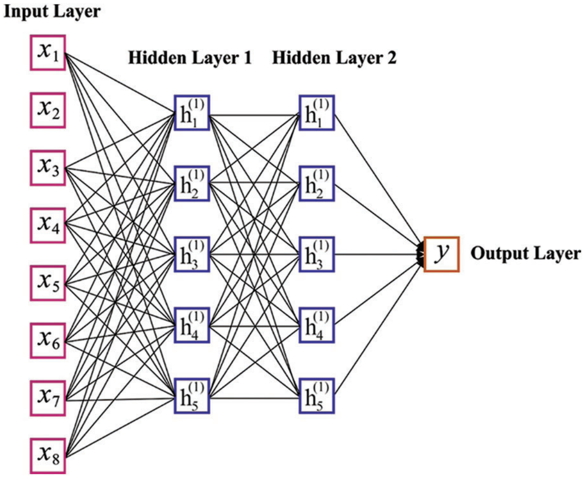

**Figure 2:** Structure of DNN

To enable the non-linear fitness capability, an activation function is higher from the hidden layer of DNN. It can utilize the sigmoid as activation function and given as follows [22].

$$S = \frac{1}{1 + e^{-x}} \tag{2}$$

An input data of network is named as x and it is activated by the mapping function $M_f$.

$$M_f = sigm(\omega_i x + \beta_i) \tag{3}$$

where ω and β imply the weight matrix and bias between the output and hidden layers respectively.

The demonstration space of hidden neuron to align with human data can establish supervised loss function of DNN. During this study, it is needed to utilize the data controlled from the data sample labels that signify the human models. To provide a conceptually labelled data instance (*x*, *l*) for hidden layer, the loss procedure is estimated as follows.

$$S(W_s,\ b_s;\ x,\ l) = \frac{1}{2m_j} \sum_{j=1}^{m} \|h_j(W_s,\ b_s;\ x) - l_j\|_2^2 \tag{4}$$

where $W_s$ and $b_s$ represents the subsets of biases and m represents the number of neurons from the hidden layer. Cross Entropy (CE) is utilized as a loss function of DNN as research to be trained and tested as well. Using the CE losses, enhances the efficiency of sigmoid and SoftMax output techniques. The CE loss is estimated in Eq. (5).

$$C_E = \frac{1}{n} \sum_{k=1}^{n} [Y_k log\hat{Y}_k + (1 - Y_k)\log(1 - \hat{Y}_k)] \tag{5}$$

where n defines the training instance quantity, $\mathrm{Y_k}$ stands for *kth* actual output of trained set, $\hat{Y}_k$ indicates the *kth* estimated output of tested set. At the same time, the loss function of the DNN model can be effectively selected by using the Gradient Descent approach.

GD is an iterative technique that aims at finding local minimal of differentiable cost function [23]. It is one of the general first order optimization algorithms from ML and DL. The GD is dependent on upgrade of all the elements in matrix $\hat{X}^{(t)}$ from the way for optimizing an objective function $J(\hat{X}^{(t)})$. A novel parameter $\mathrm{V}^{(t)}$ is given as follows.

$$\mathrm{V}^{(t)} = \alpha\nabla(J(\hat{X}^{(t)})), \tag{6}$$

$\alpha$ implies the learning rate in the range 0 and 1. $\nabla(J(\hat{X}^{(t)}))$ refers the gradient of cost function interms of parameter matrix. It is estimated as below.

$$\nabla\ (J(\hat{X}^{(t)})) = \nabla(f(\hat{X}^{(t)})) + \lambda \times \nabla(l(\hat{X}^{(t)})), \tag{7}$$

where

$$\nabla(f(\hat{X}^{(t)})) = U^{(t)} = \nabla(\|H \odot (\hat{X}^{(t)} - \mathrm{X}^{(t)})\|_F^2) = 2 \times H \odot (\hat{X}^{(t)} - \mathrm{X}^{(t)}), \tag{8}$$

$\nabla(l(\hat{X}^{(t)}))$ is computed as follows:

$$\nabla(l(\hat{X}^{(t)})) = W(t) = \nabla(\|(1 - H) \odot \hat{X}^{(t)}\|_*) = (1 - H) \odot (\hat{X}^{(t)}.\ ((\hat{X}^{(t)}).\ \hat{X}^{(t)})^{-0.5}), \tag{9}$$

It requires any sort of regularization as the inverse of square root of $(\hat{X}^{(t)})^T.\ \hat{X}^{(t)}$ may not exist, e.g.,

$$\nabla(l(\hat{X}^{(t)})) = \nabla(\|(1 - H) \odot \hat{X}^{(t)}\|_*) = (1 - H) \odot (\hat{X}^{(t)}.\ ((\hat{X}^{(t)})^T.\ \hat{X}^{(t)} + \epsilon \times \mathrm{I})^{-0.5}), \tag{10}$$

where $\epsilon$ represents the regularization parameter and $I(n \times n)$ is the identity matrix. Next

$$\nabla(J(\hat{X}^{(t)})) = 2 \times H \odot (\hat{X}^{(t)} - \mathrm{X}^{(t)}) + \lambda = 2 \times (1 - \mathrm{H}) \odot (\hat{X}^{(t)}.\ ((\hat{X}^{(t)})^T.\ \hat{X}^{(t)} + \epsilon \times \mathrm{I})^{-0.5}), \tag{11}$$

$$\mathrm{V}^{(t)} = \alpha \times (2 \times H \odot (\hat{X}^{(t)} - \mathrm{X}^{(t)}) + \lambda \times (1 - \mathrm{H}) \odot (\hat{X}^{(t)}.\ ((\hat{X}^{(t)}).\ \hat{X}^{(t)} + \epsilon \times \mathrm{I})^{-0.5})), \tag{12}$$

Utilizing the typical GD, the identified entries are well computed, it is utilized for rest of the case for estimating $\mathrm{U}^{(t)}$. For estimating $\mathrm{W}^{(t)}$, it can utilize the subsequent techniques.

### *3.3 Siamese Training Method*

One of the most complex methods for crafting the special characteristics is using numerous to single mapping for sensitive parameters. This is associated with the basic concept behindhand k anonymity, while identity is considered as a sensitive parameter. E.g., assume the problems of gender classification with feature extractors which map input images to a feature space. When they have k image of "male" class maps to k different points, an adversary would be capable of reconstructing the original image when they find out the reverse mapping functions. On the other hand, when each k distinct male image maps to an individual point in the feature space, the adversary would endure confusion in selecting the accurate identities among k potential ones. Hence, when they finetune the feature extractor in this manner where the feature of similar classes falls within a tiny neighbourhood of one another, the privacy of the input data would be maintained better. For accomplishing these tasks, they utilize Siamese framework for fine-tuning the pre-trained method on the cloud, before employing the layer partition. The major concept of finetuning with Siamese architectures is creating the depiction of semantically similar points become as closer as possible to one another when the depiction of dissimilar point falls farther from one another.

E.g., in face authentication problem, the aim is to find 2 images belongs to the same person or not, the Siamese finetuning could create any 2 images which belonging to the same person falls within a local neighbourhood in the feature space and the feature of 2 images containing mismatch face becomes farther from one another. Contrastive loss functions are employed to all pairs, thus the distance between 2 points imply minimized and when they are similar, it is maximized.

$$L(f_1, f_2) = \begin{cases} \|f_1 - f_2\|_2^2 & \textit{similar} \\ \max(0,\ \textit{margin} - \|f_1 - f_2\|_2)^2 & \textit{dissimilar} \end{cases} \tag{13}$$

where as f1 & f2 represent the mapping of data points and margin indicates a hyperparameter regulating the variances of the feature space.

They employ Siamese finetuning for increasing the privacy of feature extractors by mapping the input data to the feature space, where the sample with the similar primary class label develops closer to one another, whereas sample with distinct class labels becomes farther. It raises the accuracy of initial parameter predictions by minimizing the classification loss and simultaneously, it increases the privacy of special characteristics by minimizing the contrastive loss. The entire process of Siamese finetuning is performed in the cloud using the service providers, before employing the layer partition and providing the feature extractor model to the end user.

### *3.4 Hyperparameter Tuning Using GSO Algorithm*

Hyperparameter tuning is the process of choosing an optimum set of hyperparameters for a learning algorithm. A hyperparameter is a parameter whose value is utilized for controlling the learning process. In this study, GSO algorithm is used for the hyperparameter tuning of the DNN model. GSO is a Meta heuristic approach, offers quick convergence and fewer parameters. It is utilized for solving various optimization problems like combinatorial optimization, pattern recognition and so on. The GSO approach is inspired by the behaviour of glow worms, in detecting flaring adjacent glow worms in the search region. In the GSO approach, the attractions of the glow worms are proportion to brightness and inversely proportion to the distances between the two glow worms. Firstly, a swarm of glow worms is placed arbitrarily in the solution area. All glow worms denote the solutions of the objective function in the search area also transfer a specific number of luciferins. The level of luciferin is related to the fitness of the agents' existing location. The brighter one indicates an optimum position (an optimum solution). With a probabilistic system, all agents could be fascinated by neighbours whose luciferin intensities are greater when compared to their individual in the local decision domain and later move towards it. The densities of a glow worms' neighbour affect the decision radius and determine the local decision domain size, if the neighbour densities are lower, the local decision domain would extend for finding a higher number of neighbours, otherwise, it gets reduced to allow the swarms to split into smaller groups. The aforementioned process is continued till the algorithm fulfils the end criteria. Now, the majority of an individual collects brighter glow worm. In short, the GSO includes five major stages namely, neighbourhood select, luciferin update, decision radius update, movement and moving probability computer.

#### *3.4.1 Luciferin Update Phase*

It is based on the fitness and prior luciferin values, also the rules are provided below,

$$l_i(t+1) = (1-\rho)l_i(t) + \gamma\ \textit{Fitness}\ (x_i(t+1)). \tag{14}$$

Now, $l_i(t)$ is the luciferin value of glow worms $i$ at time $t$, $\rho$ denotes the luciferin decay constant, $\gamma$ indicates the luciferin improvement constant, $x_i(t+1) \in R^M$ signifies the place of glow worm $i$ at time $t+1$ and Fitness $(x_i(t+1))$ indicates the value of fitness of glow worms $i$ at time $t+1$.

### 3.4.2 Neighbourhood-Select Phase

Neighbour's $N_i(t)$ of glow worms $i$ at $t$ time includes the brighter one and it is expressed as follows [24].

$$N_i(t) = \{j: d_{ij}(t) < r_d^i(t); \quad l_i(t) < l_j(t)\}. \tag{15}$$

Now, $d_{ij}(t)$ represent the Euclidean distances between glow worms $i$ & $j$ at time $t$ and $r_d^i(t)$ characterizes the decision radius of glowworm $i$ at time $t$.

### 3.4.3 Moving Probability Computer Phase

The glow worms exploit the likelihood rules for moving toward other glow worms possessing high luciferin levels. The likelihood $P_{ij}(t)$ of glow worm $i$ move towards its neighbour $j$ is given by,

$$P_{ij}(t) = \frac{l_j(t) - l_i(t)}{\sum_{p \in N_i(t)} l_p(t) - l_i(t)}. \tag{16}$$

### 3.4.4 Movement Phase

Assume glow worm $i$ select glow worm $j \in N_i(t)$ using $P_{ij}(t)$, the discrete time models of the motion of glow worms $i$ is shown below,

$$x_j(t+1) = x_j(t) + step\left(\frac{x_j(t) - x_i(t)}{\|x_j(t) - x_i(t)\|}\right). \tag{17}$$

whereas $\| \cdot \|$ denotes the Euclidean norm operator and *step* represents the stepsize.

### 3.4.5 Decision Radius Update Phase

At the time of updation, the decision radius of glow worms $i$ can be represented using Eq. (18).

$$r_d^i(t+1) = \min\{r_{sy}, \text{ maxm } \{0, r_d^i(t) + \beta(n_t - |N_j(t)|)\}\}. \tag{18}$$

Now, $\beta$ denotes a constant, $r_{sy}$ is the sensory radius of glow worm $i$ and $n_t$ represents a variable for controlling the neighbours. Fig. 3 illustrates the flowchart of GSO. The algorithmic representation of the GSO algorithm is given in Algorithm 1.

**Algorithm 1:** Pseudocode of GSO Algorithm

```
Initialize $m$ dimensions
Initialize $n$ glow worms
Consider $s$ as the step size
Assume $x_i(t)$ denotes the position of glowworm $i$ at time instant $t$
Place agents arbitrarily
$deploy - agents\_randomly$;
for $i = 1$ to $n$ do $\ell_i(0) = \ell_0$
$r_d^i(0) = r_0$
assume maximal number of iterations= max_$iter$;
set $t = 1$;
while $(t \le \max\_iter)$ do:
  {
```

(Continued)

**Algorithm 1: (continued)**

```
  for all glow worms $i$ do:
    $\ell_i(t) = (1-\rho)\ l_i(t-1) + \gamma\ Fitness(x_i(t))$;
  for every glowworm $i$ do:
  {
    $N_i(t) = \{j\ :\ d_{ij}(t) < r_d^i(t);\ \ell_i(t) < \ell_j(t)\}$;
    for every glowworm $j \in N_i(t)$ do:
        $p_{ij}(t) = \frac{\ell_j(t)-\ell_i(t)}{\Sigma p \in N_i(t) \ell_p(t) - \ell_i(t)}$;
    $j = choose\_glowworm(\vec{p})$;
    $x_i(t+1) = x_i(t) + step\left(\frac{x_j(t)-x_i(t)}{\|x_j(t)-x_i(t)\|}\right)$
    $r_d^i(t+1) = \min\ \{r_{sy},\ \text{maxm}\ \{0,\ r_d^i(t) + \beta(n_t - |N_i(t)|)\}\}$;
  }
  $t \leftarrow t+1$;
}
```

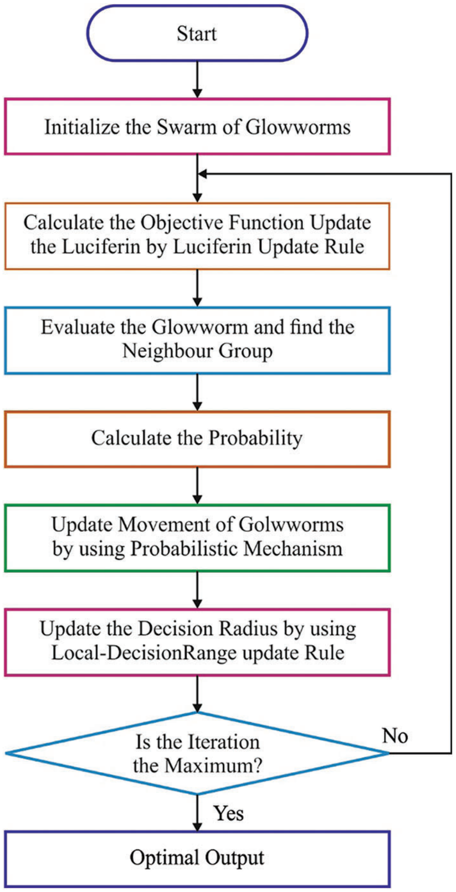
Start
Initialize the Swarm of Glowworms
Calculate the Objective Function Update the Luciferin by Luciferin Update Rule
Evaluate the Glowworm and find the Neighbour Group
Calculate the Probability
Update Movement of Golwworms by using Probabilistic Mechanism
Update the Decision Radius by using Local-DecisionRange update Rule
Is the Iteration the Maximum?
No
Yes
Optimal Output

**Figure 3:** Flowchart of GSO algorithm

## 4 Performance Validation

This section investigates the performance of the proposed model under different aspects. First, the proposed model is tested against gender classification which aims to categorize the person image into male or female with no disclosure of the identity of the persons as an instance of sensitive data. It is tested using an IMDB-Wiki dataset, comprising 62,359 images for fine tuning the gender classification model with Siamese architecture. A set of 45,000 images are utilized as training data and the remaining are used for validation process. The privacy of the measurement approaches is validated on this dataset. Besides, the performance of the proposed model is assessed by activity recognition that identifies the activity of a user from the accelerometer and gyroscope data of the smartphone. For this purpose, the Motion Sense dataset is gathered from the iPhone 6s with various actions namely downstairs, walking, upstairs and jogging.

Tab. 1 provides the accuracy analysis of the proposed model on gender classification and activity recognition tasks under various embedding approaches. The value of reduced accuracy implies better privacy. On examining the outcome of the gender classification process, the simple method has offered an accuracy of 91%, 91% and 89% on the applied data 1, 2 and 3 respectively. Besides, the reduced simple method has gained an accuracy of 83.70%, 83% and 91% on the applied data 1, 2 and 3 respectively. Moreover, the Siamese tuning approach has reached an accuracy of 88.70%, 89.70% and 87.50% on the applied data 1, 2 and 3 respectively. Furthermore, the hybrid algorithm has attained an accuracy of 87.30%, 89.90% and 87.30% on the applied data 1, 2 and 3 respectively.

**Table 1:** Predictive accuracy of different embedding methods

| Gender classification (%) | | | |
|---|---|---|---|
| Methods | Data-1 | Data-2 | Data-3 |
| Simple | 91.00 | 91.00 | 89.00 |
| Reduced simple | 83.70 | 83.00 | 91.00 |
| Siamese tuning | 88.70 | 89.70 | 87.50 |
| Hybrid model | 87.30 | 89.90 | 87.30 |
| Activity recognition (%) | | | |
| Methods | Data-1 | Data-2 | Data-3 |
| Simple | 89.00 | 86.70 | 88.20 |
| Reduced simple | 80.30 | 88.50 | 87.10 |
| Siamese tuning | 87.20 | 88.30 | 88.20 |
| Hybrid model | 85.10 | 89.80 | 89.20 |

On investigating the result on the activity recognition process, the simple approach has offered an accuracy of 89%, 86.70% and 88.20% on the applied data 1, 2 and 3 respectively. Likewise, the reduced simple approach has attained an accuracy of 80.30%, 88.50% and 87.10% on the applied data 1, 2 and 3 respectively. Followed by the Siamese tuning technique has achieved an accuracy of 87.20%, 88.30% and 88.20% on the applied data 1, 2 and 3 respectively. Additionally, the hybrid methodology has reached an accuracy of 85.10%, 89.80% and 89.20% on the applied data 1, 2 and 3 respectively.

The performance of the transfer learning model for various embeddings on distinct transitional layers for gender classification and activity recognition processes are depicted in Tab. 2. Fig. 4 investigates the face recognition accuracy of the proposed model on different sets of data. The figure shows that the simple method has obtained a higher face recognition accuracy of 23, 23 and 13 on the applied data 1, 2 and 3 respectively. In addition, the reduced simple model has gained slightly reduced face recognition outcomes with the accuracy of 3, 4 and 3 on the applied data 1, 2 and 3 respectively. At the same time, the Siamese Tuning technique has accomplished even decreased face recognition accuracy of 3, 4 and 3 on the applied data 1, 2 and 3 respectively. Finally, the hybrid model has resulted in a lower face recognition accuracy of 2, 3 and 3 on the applied data 1, 2 and 3 respectively.

**Table 2:** Gender classification and activity recognition in terms of accuracy

| Face recognition accuracy | | | |
|---|---|---|---|
| Methods | Data-1 | Data-2 | Data-3 |
| Simple | 23 | 23 | 13 |
| Reduced simple | 4 | 5 | 4 |
| Siamese tuning | 3 | 4 | 3 |
| Hybrid model | 2 | 3 | 3 |
| Gender recognition accuracy | | | |
| Methods | Data-1 | Data-2 | Data-3 |
| Simple | 95 | 90 | 79 |
| Reduced simple | 91 | 84 | 74 |
| Siamese tuning | 93 | 72 | 58 |
| Hybrid model | 91 | 70 | 52 |

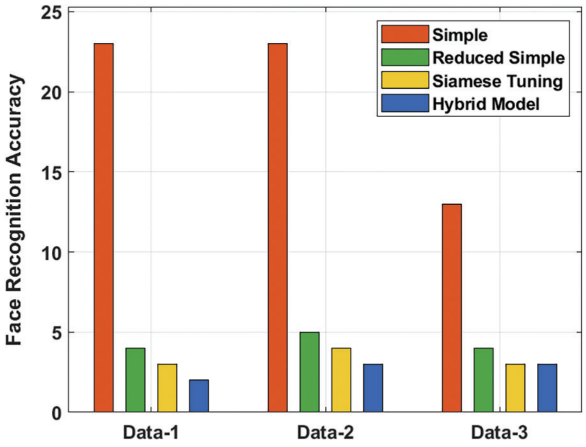

**Figure 4:** Face recognition accuracy analysis

Fig. 5 examines the gender recognition accuracy of the proposed technique on distinct sets of data. The figure shows that the simple technique has reached a maximum gender recognition accuracy of 95, 90 and 79 on the applied data 1, 2 and 3 respectively. Also, the reduced simple approach has attained reduced gender recognition with the accuracy of 91, 84 and 74 on the applied data 1, 2 and 3 respectively. Simultaneously, the Siamese Tuning approach has accomplished even reduced gender recognition accuracy of 93, 72 and 58 on the applied data 1, 2 and 3 respectively. Eventually, the hybrid approach has resulted in a lesser gender recognition accuracy of 91, 70 and 52 on the applied data 1, 2 and 3 respectively.

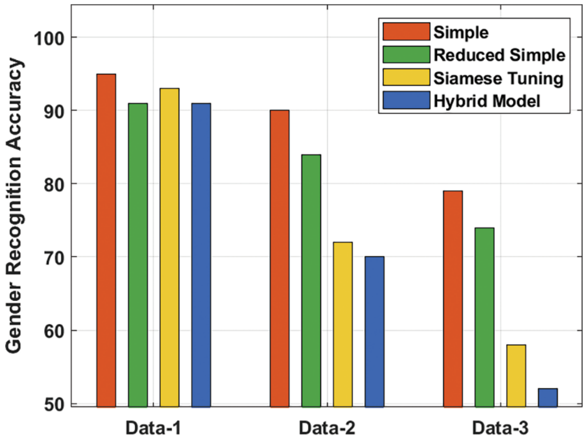

**Figure 5:** Gender recognition accuracy analysis

Fig. 6 depicts the trade-off between gender classification accuracy and identity privacy using Hybrid method. From the figure, it is shown that the gender classification accuracy with noise reduced approach is found to be rapidly decreasing with an increase in identity privacy. At the same time, the hybrid model shows considerably consistent performance with the maximum gender classification accuracy with an increase in identity privacy.

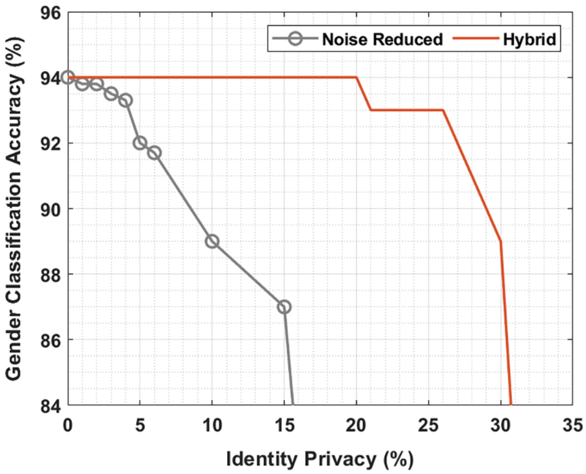

**Figure 6:** Gender classification using hybrid

Fig. 7 inspects the gender classification accuracy of the hybrid method under three different data. The figure shows that the hybrid method has shown lower accuracy on the data-1. Besides, the hybrid model has tried to showcase moderate accuracy on the data-2. However, the hybrid method has depicted maximum classification accuracy with the data-3 compared to data-1 and data-2.

Fig. 8 evaluates the trade-off between the activity recognition accuracy and identity privacy using Hybrid approach. From the figure, it is clear that the activity recognition accuracy with noise reduced approach is quickly reduced by improving the identity privacy. Followed by the hybrid method has a consistent efficiency with the maximal activity recognition accuracy by enhancing the identity privacy.

Fig. 9 scrutinizes the activity recognition accuracy of the hybrid technique under 3 varying data. The figure demonstrates that the hybrid technique has exhibited lower accuracy on the data-1. Also, the hybrid approach has tried to illustrate moderate accuracy on the data-2. But the hybrid method has the maximal classification accuracy at data-3 compared to data-1 and data-2. From the above-mentioned tables and figures, it is ensured that the proposed model has accomplished maximum privacy in the GDDC.

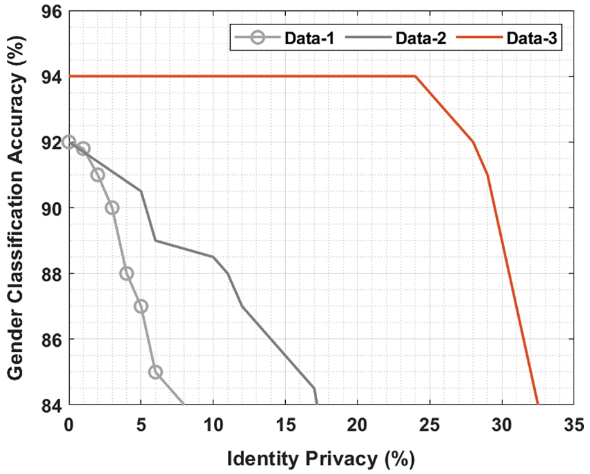

**Figure 7:** Gender classification using different data

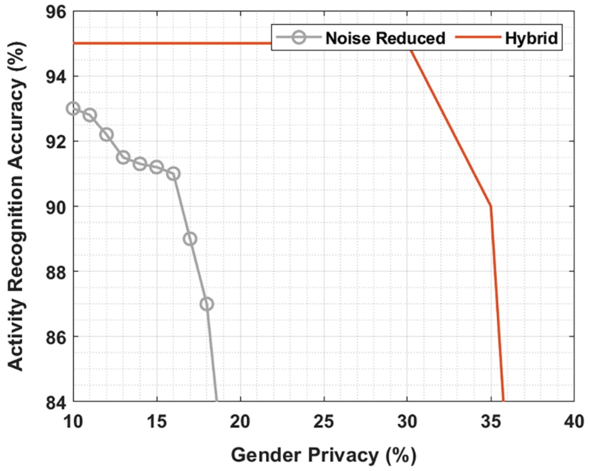

**Figure 8:** Activity recognition using hybrid

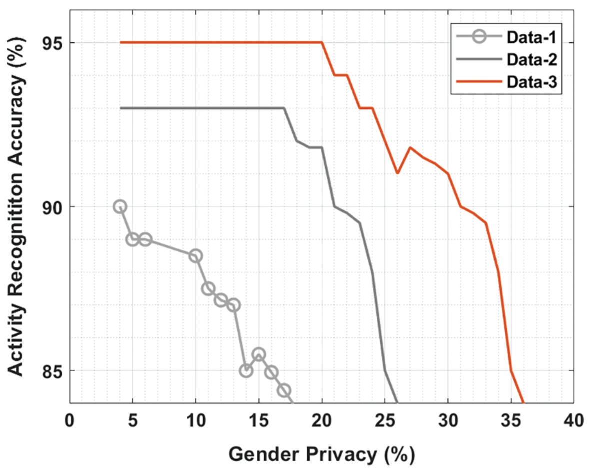

**Figure 9:** Activity recognition using different data

## 5 Conclusion

In this study, an effective hybrid framework to achieve privacy preservation in GDDC is presented. The DNN model undergoes partitioning of layers into extractor and classifier to preserve the privacy. The extractor modules execute on the master device and classifiers on the VMs. The hybrid DNN framework using GSO algorithm and sensitive data extractor acts as a core model for privacy preservation. By introducing a novel technique of sensitive extractor and tuning the data using Siamese fine tuning method, privacy can be accomplished at data loading point in the virtual machines. In addition, the DNN model is trained using gradient descent approach to appropriately select the loss function. Moreover, the use of GSO based hyperparameter optimization helps to considerably boost the overall performance. A wide range of experimental analysis is carried out and the comprehensive result analysis ensured the betterment of the proposed model over the other techniques. In future, advanced DL architectures can be used instead of DNN model to improve the privacy in geo-distributed DCs.

**Funding Statement:** The authors received no specific funding for this study.

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

## References

[1] S. Taheri, M. Goudarzi and O. Yoshie, “Learning-based power prediction for geo-distributed data centers: Weather parameter analysis,” *Journal of Big Data*, vol. 7, no. 1, pp. 1–16, 2020.

[2] N. Zainal, A. M. Zain, N. H. M. Radzi and M. R. Othman, “Glowworm swarm optimization for optimization of machining parameters,” *Journal of Intelligent Manufacturing*, vol. 27, no. 4, pp. 797–804, 2016.

[3] A. C. Zhou, Y. Gong, B. He and J. Zhai, “Efficient process mapping in geo-distributed cloud data centers,” in *Proc. of the Int. Conf. for High Performance Computing, Networking, Storage and Analysis*, St. Louis, MO, USA, pp. 1–12, 2017.

[4] C. Ramalingam, “An efficient applications cloud interoperability framework using i-anfis.” *Symmetry*, vol. 13, no. 2, pp. 218–228, 2021.

[5] S. A. Osia, A. S. Shamsabadi, S. Sajadmanesh, A. Taheri, K. Katevas *et al.,* “A hybrid deep learning architecture for privacy-preserving mobile analytics,” *IEEE Internet of Things Journal*, vol. 7, no. 5, pp. 4505–4518, 2020.

[6] K. E. Psannis, C. Stergiou and B. B. Gupta, “Advanced media-based smart big data on intelligent cloud systems,” *IEEE Transactions on Sustainable Computing*, vol. 4, no. 1, pp. 77–87, 2018.

[7] S. Divyabharathi, “Large scale optimization to minimize network traffic using mapreduce in big data applications,” in *Int. Conf. on Computation of Power, Energy Information and Communication*, Nagercoil, India, pp. 193–199, 2016.

[8] H. Torsten, J. Emmanuel and M. Guillaume, “An overview of topology mapping algorithms and techniques in high-performance computing,” in *Proc. High-Performance Computing on Complex Environments IEEE*, Haiko, Hainan, pp. 345–356, 2014.

[9] C. Stergiou, K. E. Psannis, A. P. Plageras, Y. Ishibashi, B. G. Kim *et al.,* “Algorithms for efficient digital media transmission over iot and cloud networking,” *Journal of Multimedia Information System*, vol. 5, no. 1, pp. 27–34, 2018.

[10] D. Paulraj, “An automated exploring and learning model for data prediction using balanced ca-svm,” *Journal of Ambient Intelligence and Humanized Computing, Springer*, vol. 12, no. 5, pp. 4479–4490, 2020.

[11] A. Vinothini and S. B. Priya, “Survey of machine learning methods for big data applications,” in *2017 Int. Conf. on Computational Intelligence in Data Science (ICCIDS) IEEE*, Gurugram, India, pp. 1–5, 2017.

[12] P. Subbulakshmi, “Mitigating eavesdropping by using fuzzy based mdpop-q learning approach and multilevel stackelberg game theoretic approach in wireless crn,” *Cognitive Systems Research*, vol. 52, no. 4, pp. 853–861, 2018.

[13] C. Saravana Kumar, “An authentication technique for accessing de-duplicated data from private cloud using one time password,” *International Journal of Information Security and Privacy* vol. 11, no. 2, pp. 1–10, 2017.

[14] J. Jeon, J. Kim, K. Kim, A. Mohaisen and J. K. Kim, “Privacy-preserving deep learning computation for geo-distributed medical big-data platforms,” in *2019 49th Annual IEEE/IFIP Int. Conf. on Dependable Systems and Networks IEEE*, Portland, pp. 3–4, 2019.

[15] T. Kabir and M. A. Adnan, “A scalable algorithm for multi-class support vector machine on geo-distributed datasets,” in *IEEE Int. Conf. on Big Data (Big Data)*, Dhaka, Bangladesh, pp. 637–642, 2019.

[16] Taheri, M. Goudarzi and O. Yoshie, “Learning-based power prediction for geo-distributed data centers: Weather parameter analysis,” *Journal of Big Data*, vol. 7, no. 1, pp. 1–16, 2020.

[17] T. Tang, B. Wu and G. Hu, “A hybrid learning framework for service function chaining across geo-distributed data centers,” *IEEE Access*, vol. 8, no. 12, pp. 170225–170236, 2020.

[18] G. Kaissis, A. Ziller, J. Passerat-Palmbach, T. Ryffel, D. Usynin *et al.,* “End-to-end privacy preserving deep learning on multi-institutional medical imaging,” *Nature Machine Intelligence*, vol. 3, no. 6, pp. 473–484, 2021.

[19] S. Sajid, M. Jawad, K. Hamid, M. U. Khan, S. M. Ali *et al.,* “Blockchain-based decentralized workload and energy management of geo-distributed data centers,” *Sustainable Computing: Informatics and Systems*, vol. 29, no. 5, pp. 100461, 2021.

[20] S. Nithyanantham and G. Singaravel, “Resource and cost aware glowworm mapreduce optimization based big data processing in geo distributed data center,” *Wireless Personal Communications*, vol. 120, no. 4, pp. 1–22, 2021.

[21] A. C. Zhou, Y. Xiao, Y. Gong, B. He, J. Zhai *et al.,* “Privacy regulation aware process mapping in geo-distributed cloud data centers,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 30, no. 8, pp. 1872–1888, 2019.

[22] D. Paulraj, “A gradient boosted decision tree-based sentiment classification of twitter data,” *International Journal of Wavelets, Multiresolution and Information Processing*, vol. 18, no. 4, pp. 205027-1–205027-21, 2020.

[23] W. Njima, R. Zayani, I. Ahriz, M. Terre and R. Bouallegue, “Beyond stochastic gradient descent for matrix completion based indoor localization,” *Applied Sciences*, vol. 9, no. 12, pp. 2414, 2019.

[24] R. Chithambaramani and P. Mohan, “Addressing semantics standards for cloud portability and interoperability in multi cloud environment,” *Symmetry*, vol. 3, no. 2, pp. 317–330, 2021.