Distributed Data Mining (DDM) is vital in various applications for processing large volumes of data. The datasets are saved in the local databases and operated by local communities, but it provides the solution locally and globally. However, the datasets are stored in a distributed manner which affects the scalability and reliability issues. In addition, locally stored data is influenced by security and privacy challenges. In addition, the third party may access the DDM, which causes authorization issues. Therefore, the DDM process fuses sensor data from different sources to improve knowledge discovery. During this process, the DDM faces several issues such as security concerns, privacy restrictions, technical barriers, and trust issues. To address these issues, distributed data mining (DDM) should be improved to handle homogeneous and heterogeneous data. This work uses the privacy protection-based distributed clustering (PPDC) algorithm to handle the privacy and security challenges while analyzing the distributed data. The clustering algorithm generates the semi-trusted third parties to form the cluster, which protects the data from unauthorized users. The semi-trusted party protect the locally analyzed solution by creating the random vector-based trusted process. Further, the process uses the optimized deep learning approach and clustering to improve the heterogeneous data analysis. Then the effectiveness of the introduced PPDC method is compared with existing methods, and the PPDC algorithm ensures the 0.202 error rate, 0.95 % of accuracy and manages the data security

Keywords: deep learning, privacy protection, distributed clustering, distributed data mining

UDC 004
DOI: 10.15587/1729-4061.2022.263692

PRIVACY PROTECTION BASED DISTRIBUTED CLUSTERING WITH DEEP LEARNING ALGORITHM FOR DISTRIBUTED DATA MINING

Alaa Thamer Mahmood
Master of Information Technology, Assistant Lecturer
Middle Technical University
Technical Instructors Training Institute
Zafaraniya str., 961, Baghdad, Iraq, 00964
Raed Kamil Naser
Corresponding author
Computer Officer, Military Training Directorate
Ministry of Defense, Iraq
Zafaraniya str., 961, Baghdad, Iraq, 00964
E-mail: raed_kamil_naser@student.usm.my
Sura Khalil Abd
Doctor of Network and Communication
Systems Engineering, Lecturer
Department of Computer Techniques Engineering
Dijlah University College
AlMasafi, Baghdad, Iraq, 00964

Received date 24.06.2022
Accepted date 23.08.2022
Published date 31.08.2022

How to Cite: *Mahmood, A. T., Kamil Naser, R., Khalil Abd, S. (2022). Privacy protection based distributed clustering with deep learning algorithm for distributed data mining. Eastern-European Journal of Enterprise Technologies, 4 (9 (118)), 48–58. doi: https://doi.org/10.15587/1729-4061.2022.263692*

1. Introduction

A Knowledge Discovery Database (KDD) [1] is widely utilized in various applications for processing and identifying the knowledge from the gathered data. The KDD process utilizes data mining techniques, which consist of data collection, selection, cleansing, prior knowledge extraction and data interpretation [2]. The KDD is used in different applications like fraud detection, marketing, manufacturing and telecommunication. Improving data mining in different fields causes distributed data mining (DDM) [3]. The DDM can analyze a large volume of data concerning the physical location, storage, computing units and sources. One general distributed environment is the Internet, where a large volume of data is stored. The DDM resolves the single-site centralized storage issues [4]. In addition, the distributed mining procedure gives global insights for data that helps to generate the global pattern for a specific problem, and the generated solutions are transferred globally, reducing the expensive computations like storage and cost [5]. The DDM process can transmit a large volume of data sites to other parts, which helps maintain scalability and improves performance in specific applications. However, the DDM process has a few challenges while handling the heterogeneous/homogeneous data, data fragmentations, data replication, and communication cost resulting in integration and data skewness [6]. These difficulties affect the complete data mining performance. Therefore, the distributed algorithm should be implemented by considering these research issues.

The distributed data mining algorithm [7] should create towards the centralized collected data and must analyze the data distributed. Suppose the dataset is high dimensional, and the data mining technique consumes high computation time and speed [8, 9]. Then, the parallel knowledge analysis approaches are utilized to resolve the problem with maximum performance using multi-computer devices. Therefore, machine learning techniques [10, 11] are applied to investigate the distributed dataset for exploring the results. Among the various techniques, clustering is an effective approach that analyses similar data to identify the unknown results. The clustering process [12, 13] identifies the similarities between data and the similar data allocated to the same cluster.

The allocated data are denoted as the attributed, which are done according to the density, centroid, hierarchical and spectral. However, the clustering may have huge complexity regarding privacy and security issues in data.

In DDM, the data is analyzed in terms of vertical and horizontal aspects or heterogeneous and homogeneous perspectives [3]. The existing clustering algorithm deals with the homogenous data to solve the distributed clustering issues. Later, heterogeneous data are investigated to analyze the distributed data in a difficult situation. The heterogeneous dataset has a large dimension that requires additional effort to minimize the computation complexity [14] while maintaining the relevant data. During this process, the system combines the data from different resources to make an effective knowledge discovery process. The data has been distributed in various locations with sensitive data. Hence the fusing process needs to manage the data security and privacy factors. Therefore, an effective distributed data mining system should be implemented to handle the large volume of data with minimum computation complexity, including security and data dimensionality issues. Many researchers use correlation and principle component analysis to overcome this issue to minimize the data dimensionality. This process has non-linear interactive issues, which avoids the few important data while forming clustering. However, the existing methods attain high error rates while clustering similar information. The issues are overcome by applying the mayfly optimized deep learning algorithm to learn the data features and handle the data replication problem. In addition, the neural network uses the local dataset information to identify the global patterns for specific problems. The global patterns are obtained from previous learning and feature relationship. Then the network uses the optimization algorithm to update the network parameter, minimizing the optimization problem and false data identification rate.

Distributed data mining is a term that describes the process of extracting information from several locations at once. Using a computer network, local computers store the data sets in local databases, which are accessible over the network. Machine learning may be used in a variety of ways in the field of security, including malware analysis, prediction, and clustering of security events. It may be used to identify previously discovered attacks that have no known signature. Unlabeled datasets may be clustered or clustering analyzed using a machine learning approach. It may be described as a means of arranging data points into various clusters consisting of related data points. Researchers can better detect malware in encrypted traffic and identify insider threats by using machine learning to analyze data and learn patterns. This can predict where people will be browsing in bad neighbourhoods on the internet and protect their personal information by identifying unusual behaviour.

Further, the distributed data mining, k parties, has large datasets needed to form the clusters without compromising security and privacy. In DDM, the dataset is distributed in a different environment that different users have accessed. Therefore, the trusted party should create to manage the heterogeneous data. Then the main research devoted is to introducing optimized deep learning with a privacy protection-based distributed clustering algorithm for managing the data security and privacy factors. The algorithm generates the semi-trust party to manage the entire local pattern for generating the global solutions. The optimized machine learning algorithm based on created clusters is more useful for securely analyzing data.

2. Literature review and problem statement

In [15] discusses the distributed data privacy managing process in IoT applications. This work intends to manage the privacy, security and sensitive data target attacks. Initially, data were analyzed to extract sensitive information without eliminating the privacy criteria. Risk assessment, privacy valuation, private data trading, and sensitive data should be considered during this process to manage data privacy. These considerations are more helpful in managing privacy in IoT applications. However, this system requires the additional efforts while managing the sensitive data.

In [16], federated and machine learning techniques were introduced to manage privacy in a distributed environment. This work intends to improve data privacy by utilizing available resources. The author uses the distributed perturbation algorithm (DISTPAB) to investigate and partition data horizontally. Asymmetry resources are utilized during this process to eliminate the bottleneck problem with minimum computation complexity. The continuous utilization of the federated learning algorithm maximizes system scalability, reliability, accuracy, efficiency and high attack resistance compared to existing methods. In addition this, system requires the improved encryption algorithms to reducing the intermediate attacks.

In [17], implementing the hybrid security model improves privacy in DDM. The introduced method uses the *k*-means clustering and naïve Bayes classification method to ensure the two levels of security. First, a four-dimensional rotation transformation algorithm is applied to transfer the data to a non-understandable format. Then secure summation protocol manages the security while distributing the data. In addition, a machine learning algorithm was applied to verify sensitive data security. Thus the system ensures the high accuracy of security in the DDM environment. However, this process requires the optimization techniques to handling the sensitive data.

In [18], maintaining privacy in distributed data environment using the *K*-means clustering. This system aims to manage data privacy in a distributed environment. The *K*-means clustering approach is applied to manage the local differential privacy by highly perturbed data. Then local privacy is further enhanced in every user using the effective cluster formation. For every iteration, the results are examined with the privacy requirement to the introduced system guarantee security and privacy with qualitative analysis.

In [19], managing data privacy in an insecure environment using the privacy preserved *K*-means clustering approach. The distributed clustering algorithm uses the elliptic curve cryptography approaches to predict the external adversaries. The successful prediction of each attack helps ensure security and privacy with minimum computation cost and complexity.

In [20] it is introduced that the privacy data availability clustering approach to enhance the security and privacy in intelligent electrical services. This work aims to reduce privacy disclosure and data distortion issues using the data clustering algorithm. The algorithm uses the K-means algorithm to compute the distance between the data and centroid computation. According to the values, outliers are eliminated from the dataset, which helps to minimize unauthorized activities. Thus the introduced privacy data clustering algorithm ensures the security and privacy of IoT electrical services with minimum computation complexity.

In [21], the blockchain-based privacy algorithm in a distributed database (DEPLSEST) predicted user behaviour and privacy factors in social networks. The introduced blockchain with clustering algorithm uses synchronization operations to manage the privacy in local database storage. Then ledgers are utilized to maintain every action by using the consensus protocol. The protocol gives effective byzantine fault tolerance compared to the other secure clustering algorithm.

In [22] it is introduced that the multi-core DBSCAN approach for managing data privacy for network usage data. This process gives security and privacy to data without requiring the adversaries' prior training or knowledge. The sensitive data has been protected by including the random noise in the transmitted data. After that, similar information is clustered based on the DBSCAN, which eliminates the privacy leakage by including the Laplace noise. The effective inclusion of these two noises is difficult to predict by third parties. Therefore, the introduced method ensures the security and privacy of sensitive data with minimum computation complexity.

In [23] detected, intrusion in VANET using the distributed privacy-preserving collaborative algorithm. This work aims to reduce the malicious node involvement in the distributed data. The privacy-preserving approach uses alternating directions to reduce the empirical risk minimization problem and maximize the intrusion detection accuracy. In addition, a dual-variable perturbation procedure is applied to manage dynamic privacy. According to the various researcher's opinions, data privacy and security is established in DDM by applying different clustering and machine learning algorithm.

Based on the above survey, there are several challenges in existing methods in achieving high accuracy and reducing the error rate. However, those methods consume high computation complexity and less security while handling the heterogeneous data in the distributed environment. Then the existing systems are fails to handling the high-dimensionality related multiple resource data. The high -dimensionality leads to minimize the overall clustering process and causes to privacy issues. This causes to create the deviation between the actual and computed values while clustering and user authentication process. To overcome this issue, privacy protection-based distributed clustering (PPDC) with optimized deep learning approaches is included in this work. The introduced PPDC-ODL method performance is compared with the existing researcher's works such as [16, 17, 21, 22]. Among the several methods, these methods are chosen because of effective guidelines and methods utilized to preserve the sensitive data in the distributed environment. The successful utilization of these algorithms receives reasonable accuracy with minimum computation complexity.

3. The aim and objectives of the study

The main aim of this study is to overcome the difficulties involved in the high-dimensional data clustering process. The existing systems are fails to analyze and corrupted data in the database which leads to reduce the security and privacy issue. In addition the high-dimensional data consumes more computation time and security is one of the major challenges.

To achieve this aim, the following objectives are accomplished:

– maximizing the security and privacy factor while clustering the data from multiple location by applying distributed clustering process. The clustering process done according to the user preference which helps to verify and identifying the user that improves the overall security;

– to reducing the difficulties in high-dimensionality data handling issues and minimizing the deviation between the sensitive data security creation;

– to reduce the deviation between the actual and computed data while grouping the data multiple resources. The deep learning model uses the network parameters that continuously update, minimizing deviations.

4. Materials and methods

The study aims to manage data privacy and security because the system uses a large volume of data from various locations for analyze. The system's objective is achieved with the help of the privacy protection-based distribution clustering (PPDC) algorithm. The PPDC method investigates the semi-trusted third parties to cluster the data according to the user requirement. During the analysis, an optimized deep learning network approach is applied with the clustering process to enhance the overall data privacy protection.

Measures to keep sensitive information out of the hands of unauthorized parties are called «confidentiality» measures. It is standard practice to classify data based on the potential harm it may do if it comes into the hands of the wrong people. Data clustering is primarily concerned with identifying groups of related objects within a dataset. Data sets for clustering might be held by a single entity, or they can be enriched by pooling information from several databases to increase the clustering effort. This study uses cryptographic techniques like homomorphic encryption to help us achieve this goal, and let's also take advantage of the distributed system structure and boost computation and communication efficiency through data packing to further advance our current state-of-the-art in processing encrypted data. To recognize the irrelevant activities involved in the DDM clustering and machine learning techniques are uses. During the clustering process, assume every data has been sensitive data. Therefore, the introduced clustering process establishes the privacy and security. The clustering algorithm generates the global solutions from the local dataset by generating the semi-trusted third party. The clustering process examines each local dataset from the large dataset (whole dataset). The global solution or representations are formed from the local datasets used to form the global clustering. The formed global clusters are applied in the local datasets to determine the effectiveness of the distributed clustering. The successful utilization of these local and global representations is more useful in solving the clustering problem with minimum computational complexity. Initially, the local datasets are split into homogeneous (horizontal) and heterogeneous (vertical) distributed datasets. The distributed datasets have the same attributes in all datasets, but the attributes vary in heterogeneous situations. During the clustering process, the similarity between the data has been examined to reduce the irrelevant data involvement. At the time, the clustering problem had to be minimized to improve the overall system efficiency. Then the overall structure of the distributed clustering is illustrated in Fig. 1. Security measures should be implemented to restrict data at the greatest possible level. Moderate risk to the Institution or its affiliates might be associated with the unauthorized disclosure, modification, or destruction of material categorized as private. For the purposes of maintaining

data integrity, confidentiality, and availability, information security refers to the process of guarding digital data and information systems against illegal access, use or disclosure.

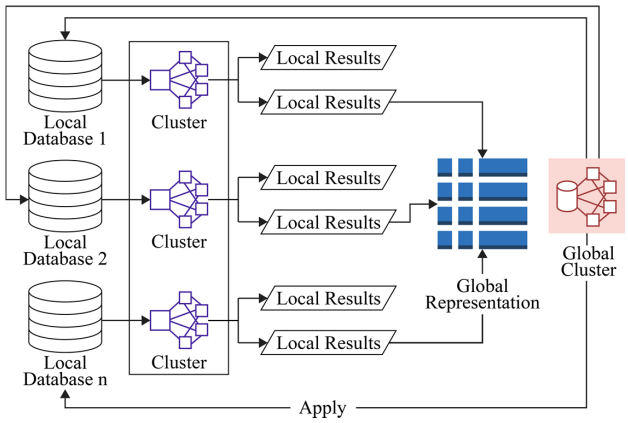

Fig. 1. Distributed data mining clustering in heterogeneous database

Fig. 1 illustrates the heterogeneous database-based clustering process in DDM. The local database information is clustered by applying the introduced cluster algorithm; the local results are examined in the global representation using global clustering. The obtained cluster value is again propagated to the local database to ensure security and privacy and improve the overall clustering accuracy. In this work, a cryptographic protocol that is a privacy-preserving process is applied along with the *K*-means clustering to improve security. The main intention of the work is to manage the private or sensitive data from the service provider. For every clustering, the cluster centroid. C_k is selected for each preference vector P_i. The clustering process has the following roles in improving the clustering efficiency. The data clustering follows the cryptographic protocol and ensures the data privacy. The service provider collects the data and forms the dataset, which gives data to the user request in the distributed environment. The data is accessed according to user preferences and similar information. An information security system safeguards sensitive data from unauthorized access and use. This includes preventing unauthorized viewing, alteration, recording, and destruction. Customer's personal and financial information, along with intellectual property, must be protected from unauthorized access. While information security refers to the protection of data stored, processed, and communicated in accordance with the functions and goals of an organization's information systems, information privacy refers to the protection of data relating to a subject's identity. Security of information.

It is considered that every data has public key pair with respective certificates, and the data provider identifies the K points in the initial cluster centroids. For every data, the clusters are formed using the following steps:

– the service provider generates the local dataset *M* group of data, and each group G_m; $m=\{1, 2, ... M\}$. Each local dataset group select the data randomly H_m, and the total number of data N_d in the group is computed as $N_d=N_g \cdot M$;

– then, in every iteration, the service provider encrypts the data in the group G_m by using the public key H_m;

– the encrypted cluster centroids are sent to the local dataset users and transmitted to the service provider. The encrypted cluster centre is more useful for obtaining the data in the distributed data;

– for every data, service provider contact with the public key and the closest centroid value is selected. According to the centroid value, data has been selected, and clusters are formed;

– the service provider aggregates the inputs and interacts with the public key to obtain the clustering results.

Then the detailed privacy protection-based clustering process is explained to maintain the security and privacy of distributed data mining environment.

The first step of the clustering process is choosing the cluster centroid (K-points) in the R-dimensional space. The service provider generates the *M* groups that consist of N_g of data. The service provider selects the random data in current iterations for every iteration. The helper in the middle is used to cluster the data to form the local dataset in the whole dataset. Then the generated public key is transmitted to the user for accessing the data. The successful generation of the key is used to reduce the intermediate access. There are two different keys that are used in public key encryption, known as asymmetric encryption or public key encryption. Using public key encryption is a vital part of Internet security technology. The web server stores two paired keys that may be used to encrypt and decode information in public key encryption. Every user that tries a secure connection with the server will get a copy of the public key, while only the server retains the server's own private key. The encrypting process ensures the data security and confidentiality. The encryption process hiding the sensitive data by transferring original content into the encrypted format that manages the data confidentiality.

The service provider should compute the distance between each data point in each G_m group. The Euclidean distance is computed for every preference user data P_i and cluster centroid C_k.

$$D^2_{(i,k)} = P_i - C^2_k = \\ = \sum\nolimits_{r=1}^{R} p^2_{(i,r)} + \sum_{r=1}^{R}\left(-2p_{(i,r)}c_{(k,r)}\right) + \sum_{r=1}^{R} c^2_{(k,r)}. \quad (1)$$

In (1), *i* user has user preference vector P_i, and the service provider holds the cluster location C_k. The computed locations should be stored in terms of privacy and security to eliminate third-party access. During this process, the distance between the data point should be computed without losing security and privacy. Therefore, for every user *i* the homomorphic property should be maintained for cluster centre C_k encryption. Then, the service provider adds the cluster centre $\sum_{r=1}^{R} c^2_{(k,r)}$. The encrypted data distance should be estimated using (2):

$$\left[\left[D^2_{(i,k)}\right]\right]_H = \left[\left[\sum\nolimits_{r=1}^{R} p^2_{(i,r)}\right]\right]_H \times \\ \times \prod_{r=1}^{R}\left[\left[c_{(k,r)}\right]\right]^{-2p_{(i,r)}}_H \cdot \left[\left[\sum_{r=1}^{R} c^2_{(k,r)}\right]\right]_H. \quad (2)$$

In (2), the distance between the user preference data should be computed according to the homomorphic encryption property. This computation helps to minimize the computational overhead issue for user *i* data. During the computations, $K(R+1)$ multiplication and KR exponentiation are utilized for every data and every user *i* by mod n^2 in the distributed data environment. The computational expense is minimized by taking the Euclidean distance value for every

data i. e., a single packet value. Therefore, the service provider computes the distance value as a packed set defined in (3).

$$\left.\begin{aligned}\tilde{C}_1 &= c_{(1,1)}c_{(2,1)}\ldots\ldots c_{(K,1)},\\ \tilde{C}_2 &= c_{(1,2)}c_{(2,2)}\ldots\ldots c_{(K,2)},\\ &\ldots.\\ &\ldots.\\ &\ldots.\\ \tilde{C}_R &= c_{(1,R)}c_{(2,R)}\ldots\ldots c_{(K,R)}.\end{aligned}\right\} \quad (3)$$

In (3), the ... is denoted as the concatenation and the simplify the clustering centre computation as (4):

$$\tilde{C}_r = \sum_{k=1}^{K} C_{(k,r)} \cdot \left(2^l\right)^{k-1};\ \text{for } r \in \{1,2,\ldots.R\}. \quad (4)$$

In (4), the packed dataset centroid values are computed with the size of l. The length is computed in the R dimensional model and bit length w of centroid value. The distance is estimated for the R size of $2w$ bits. Along with this, the service provider computes the packed centroid value according to (5):

$$\tilde{C}^2 = \sum_{r=1}^{R} c_{(1,r)}^2 \sum_{r=1}^{R} c_{(2,r)}^2 \sum_{r=1}^{R} \ldots\ldots \sum_{r=1}^{R} c_{(k,r)}^2. \quad (5)$$

In (5), the computed centroid values are encrypted by the service provider with public key H_m, and the encrypted details are sent to the user for accessing the distributed data in a secured manner. Then, every user (i) in G_m estimate the packet distance to group the similar details, which is done according to (6):

$$\left[\left[\overset{\vee}{D_i^2}\right]\right]_H = \left[\left[\overset{\vee}{C^2}\right]\right]_H \cdot \prod_{r=1}^{R} \left[\left[\overset{\vee}{C_r}\right]\right]_H^{-2p_{(i,r)}} \cdot \left[\left[\overset{\vee}{P^2}\right]\right]_H. \quad (6)$$

In (6) $\left[\left[\overset{\vee}{D_i^2}\right]\right]_H$ is the distance between the cluster centroid $\overset{\vee}{C_r}$ and the data points in the database, where,

$$\left[\left[\overset{\vee}{D_i^2}\right]\right] = \left[\left[D_{(i,1)}^2 D_{(i,2)}^2 \ldots. D_{(i,K)}^2\right]\right]_H,$$

and

$$\left[\overset{\vee}{P^2}\right] = \sum_{r=1}^{R} p_{(i,r)}^2 \cdots \sum_{r=1}^{R} p_{(i,r)}^2 \sum_{r=1}^{R} p_{(i,r)}^2.$$

The computed distance $\left[\left[\overset{\vee}{D_i^2}\right]\right]_H$ is transferred to the service provider. The estimated distance value should be packet after performing the encryption value. The packed information requires the R+1 multiplication and R+1 encryption process. After computing the distance measure, the closest cluster data has to be computed. When it comes to protecting data and privacy service tasks, companies may benefit from a variety of cloud-based or web-delivered services that allow them to secure their data assets while also giving them the ability to enhance their network security and recovery choices.

Next, the closest cluster should be identified according to the computed $\left[\left[\overset{\vee}{D_i^2}\right]\right]_H$ value. Here, the service provider with public key H_m to identify the minimum distance. The minimum distance data are considered as the closest cluster. The computed packet distance is transmitted to the user with the decryption key to obtain the cypher text. After performing the decryption process, the computed values are unpacked and generate the vectors such as $\gamma_{ik,}$ which is equal to 1 when the computed distance value is minimum. The minimum distance data is grouped to gather and form the cluster. Then the cluster centroid should be updated for every encryption. For the centroid updating, the service provider computes the user preference in each cluster that is defined in (7):

$$\left[\left[\overset{\vee}{S_{(i,r)}}\right]\right]_H = \left[\overset{\vee}{\Gamma_i}\right]_H^{p_{(i,r)}} = \left[\gamma_{(i,1)} \cdot p_{(i,r)} \ldots. \gamma_{(i,K)} \cdot p_{(i,r)}\right]_H. \quad (7)$$

In (7), the encryption process has the vector $\gamma_{(i,1)} \cdot p_{(i,r)}$ multiplication for every compartment of K packet value. The closest cluster has only the $p_{(i,r)}$ value for every cluster because it has zero $K-1$ compartment value. Finally, the computed preference $\overset{\vee}{S}_{(i,r)}$ value is sent to the service provider; here $r \in \{1, 2, 3, \ldots.R\}$. The cluster centroid value is updated according to the service provider's prescribed details like the number of users and preference. For one compartment in the centroid, packets $\left[\overset{\vee}{\Gamma_i}\right]_H^{p_{(i,r)}}$ are available in a service provider that helps to calculate the number of users in the group while performing encryption. After updating the cluster centre, user preference should be summed up by the service provider for every cluster:

$$\left[\left[\overset{\vee}{P_{\Sigma_{(m,r)}}}\right]\right]_H = \prod_{i \in G_m} \left[\left[\overset{\vee}{S_{(i,r)}}\right]\right]_H = \left[\left[\sum_{i \in G_m} \overset{\vee}{S_{(i,r)}}\right]\right]_H. \quad (8)$$

The computed preference values are encrypted, ensuring security while transmitting and accessing data in the distributed environment. Then the encrypted information is transmitted in the group, and the data is accessed by performing the decryption process. The new cluster centroids are computed to form the cluster to capture the record with minimum effort.

This work introduces an optimized deep learning model to analyze the heterogeneous data during the formed cluster. The deep learning approach reduces the clustering problem while investigating similar data. The deep neural network minimizes the dimensionality of data, improving the overall data analysis process.

The work's main objective here is to reduce the loss function while analyzing the heterogeneous data. The deviation or loss value is minimized by updating or optimizing the network weight value. The loss objective function is defined in (9):

$$J(x,\hat{x}) = x - \hat{x}^2 = x - \sigma'\left(W'\left(\sigma\left(Wx+b\right)\right)+b'^2\right). \quad (9)$$

In (9), the loss value J is computed from input x and output $\hat{x}$. The $\hat{x}$ is estimated from the latent variables such as weight (W, W'), bias (b, b') and activation function σ. The network has two parts an encoder and a decoder. The output is computed by identifying the relationship between the data and variables used to improve the overall heterogeneous data analysis.

The network uses both encoder and decoder processes in local data clustering to form the clusters. First, a bottleneck scheme is applied in the encoder that compresses the heterogeneous features that help to reduce the data dimensionality. The dimensionality of the data helps to improve the overall data clustering accuracy. The network uses the hidden layer according to the number of inputs which segments the data accordingly. The distributed and heterogeneous dataset has replicated data that affect the computations and increase complexity. Therefore the distributed data replicated data has to be analyzed and concatenated to reduce the further global representation analysis difficulties. Several local datasets are investigated using a deep learning approach to form the global clusters. The distributed dataset has n local dataset, which is processed by a deep network concurrently, and dataset features are investigated in each layer. The extracted local dataset features are grouped using the deep learning networks to obtain the global features. The obtained global clustering information is trained by a network that extracts the global code that depicts the local code. The deep learning, hidden layer combines all replication data and features are investigated in this layer to eliminate the dimensionality issues. Then the local and global clustering results are obtained using (10):

$$J\left(X_i, X_i'\right) = X_i - X_i'^2; \ \forall i = 1,2,\ldots.n. \tag{10}$$

As said earlier, the local datasets results of the hidden layer should be concatenated to get the global results; the concatenate result is described in (11):

$$J\left(\text{concat}\left(H_{jk}\right), H_{jk}'\right) = \text{concat}\left(H_{jk}\right) - H_{jk}'^2,$$
$$\forall j = 1,2,\ldots.n. \tag{11}$$

In (11), the replica data input is represented as X_i, is the reconstructed output values of replica data, j-th replica data's local code is denoted as H_{jk}, concatenate function is defined as concat (.). The global input matrix is defined as concat (H_{jk}), and the reconstructed global input matrix is defined as H_{jk}. The effective computation of the distributed data in the hidden layer successfully reduces the replicated data, reducing the data dimensionality and minimizing the computation complexity. Here the computed global pattern should be compared with the trained pattern to evaluate the difference between the outputs. According to the computations, the network variables should be updated to minimize deviations.

Optimization algorithms are applied to train the features to overcome the gradient descent and computational efficiency problems. To overcome these issues, mini-batch processing with stochastic gradient descent is applied to minimize the deviation between the output. The stochastic algorithm uses the first-optimization process that improves the network learning process. During the training, the network parameter θ should be updated as defined in (12):

$$\theta = \theta - \eta \cdot \nabla_\theta J\left(\theta; x^{(i;i+k)}; y^{(i;i+k)}\right). \tag{12}$$

In (12), the gradient is denoted as ∇_θ, the learning rate is η, and k is having as $1<k<n$. The training samples are selected from the entire dataset to reduce the computation complexity, such as computation cost. The training samples are examined to get unbiased data using the stochastic process, and momentum learning is applied to improve the data analysis process. The momentum optimization is done by using (13):

$$\left.\begin{aligned} v_i^{(t+1)} &= \alpha v_i^{(t)} - \eta \nabla_\theta J(\theta), \\ \theta_i^{(t+1)} &= \theta_i^{(t)} + v_i^{(t+1)}. \end{aligned}\right\} \tag{13}$$

In (13), the momentum parameter belongs to the value (0, 1), i^{th} parameter velocity is denoted as v_i, and the learning parameter is η. As said, the momentum optimizer helps to resolve the gradient problem in the hessian matrix analysis. However, the input and output deviation should be minimized further using the optimized network parameters. In this work, a genetic bee optimization algorithm is applied, which effectively regularizes the network performance and minimizes the binary optimization issues. Here the algorithm uses the employee, onlooker and scout bee characteristics, selection, crossover and mutation operators to select the best network parameters. The network parameter should be identified in the population size PS with a limited L parameter. In each layer, the populations are analyzed, and the parameter is computed using (14):

$$u = u_{i,j}^{\min} + \text{rand}[0,1]\left(u_{i,j}^{\max} - u_{i,j}^{\min}\right). \tag{14}$$

In (14), the random value is selected from 0, and 1, and the decision is taken in the maximum and minimum limits of the upper boundary. Here, the maximum objective function is utilized to select the optimized parameter and the right value is selected according to the food searching process.

Then the velocity of the searching process is defined using (15):

$$v_{i,j} = u_{i,j} + \theta_{i,j}\left(u_{i,j} - u_{k,j}\right). \tag{15}$$

The employee bee chose the solution $v_{i,j}$, which is analyzed by the onlooker function. In addition to this, the probability value of the selected solution is estimated to identify the best network parameter defined in (16):

$$p_i = \frac{fit_i}{\sum_{j=1}^{PS} fit_j}. \tag{16}$$

Finally, the selected solution was investigated using a genetic operator, which minimizes the binary optimization problem of the network parameter updating process (17):

$$U_i: i = 1,2,\ldots,SN; \ u_{ij} = \begin{cases} 0 \text{ if } G(0,1) \le 0.5, \\ 1 \text{ if } G(0,1) > 0.5. \end{cases} \tag{17}$$

The solutions are predicted using the uniform distributed value. According to the above process, the network uses the encoder and decoder to effectively investigate the local data and replicate information. The concatenated replicated data reduces the local dataset dimensionality issue with minimum complexity. In addition, the reconstructed information is more useful for solving the optimization problem while forming the clustering. The reconstructed local dataset details are investigated by privacy preserved clustering that forms similar information and maintains security. The privacy-based formed local clusters are more useful to get the global code, improving the results on various applications. Then the effectiveness of the system is evaluated using the experimental analysis.

5. Results of privacy protection based distributed clustering (PPDC) with optimized deep learning approach (PPDC-ODL)

5. 1. Consistent Accuracy Index (CAI)

This section evaluates the effectiveness of the introduced privacy protection-based distributed clustering (PPDC) with an optimized deep learning approach (PPDC-ODL)-heterogeneous data clustering process in DDM. The system's excellence evaluated how similarly the data are clustered using the distributed datasets.

The formed clusters' efficiency is determined using the Consistent Accuracy Index (CAI), which is computed using (18):

$$CAI = \frac{\sum_{i=1}^{k}\sum_{j=1}^{n_j} I\left\{m^C\left(c_{ij}\right) - m^D\left(c_{ij}\right)\right\}}{\sum_{i=1}^{k} n_i}. \quad (18)$$

In (9), total local data is denoted as n_i, indicator variable is I, which belongs to {1, 0}. The centralized and distributed clustering algorithm outputs are represented as $m^c(c_{ij})$ and $m^D(c_{ij})$. If the computed clustering output of $m^c(c_{ij})$ and $m^D(c_{ij})$ is equal, it has 1 else 0. Let's consider the CAI value is 1 then, $m^c(c_{ij})$ and $m^D(c_{ij})$ values are the same; if CAI value is 0 then $m^c(c_{ij})$ and $m^D(c_{ij})$ have different results.

In this work, three datasets such as Mnist [24], Covertype [25] and Sensorless dive diagnosis dataset (SDDD) datasets [26], are utilized to evaluate the introduced system efficiency. This dataset information is obtained, and CAI values are computed to determine how effectively the introduced system resolves the clustering problem with minimum computation complexity. The Mnist dataset has 70,000 digits handwritten from 0 to 9; 785 features are presented in the dataset. The Covertype database has 581,012 instances with 54 features; it is used to identify the forest cover type. The SDDD dataset has 58,509 instances and 49 features extracted from the electric current signals. These datasets information is collected from the UCI machine repository, and the features are used to cluster the data, help to improve the overall training process. The detailed description of the datasets is illustrated in Table 1.

Table 1

Dataset Description

Name of the datasets	Covertype	Mnist	SDDD
Instances	581,012	70,000	58509
Classes	7	10	11
Replica data	2	4	3
Replica (Attributes)	10/44	196/196/196/196	16/16/16

Table 1 illustrates the attributes in the three datasets. The collected central dataset was divided into the local datasets to evaluate the effectiveness of the deep learning-based clustering process. The collected datasets are split into a local database for analyzing the introduced privacy preserved clustering method effectiveness. Consider the Mnist dataset, which has 4 replica information, and 196 has 4 entries. The repeated attributes create confusion and affect the system performance in the distributed environment. The dataset has different features which are continuous, and most of the variables are dummy, but the local dataset features are distinctive. Confidentiality is the protection of data against unexpected, illegal, or unauthorized access, disclosure, or theft. It's all about protecting the privacy of information, including permissions to access and use it. Encryption is one of the most effective techniques to ensure the privacy of data. The proposed algorithm is used to transform data into an unreadable format during the encryption process. This information can only be accessed by those with the proper credentials. Encrypted data may be deciphered by others. The deep learning network uses the inputs from the database and produces the replica reconstructed output by processing in the hidden layer. Here, the seven hidden layers with 10 neurons are used for each replica to compute the reconstructed output. During the computation, the network uses the hyperbolic tangent as an activation function with 0.2 as the dropout rate and 100 mini-batch sizes. The local codes are obtained for every distributed dataset by running the attributes in a deep learning network. Optimized global codes are obtained from the local code by forming the cluster using privacy preservation. The encoder-decoder procedure uses effective learning patterns that minimize the network dimensionality and minimum computation complexity. Here, the computed semi-trust third-party process ensures the data's security and privacy, and the clusters are formed by effective computation of data similarity. The generated local and global code effectiveness is evaluated using the CAI metric. The obtained results are compared with the existing researcher's techniques such as distributed perturbation algorithm (DISTPAB) [16], k-means clustering and naïve Bayes classification (K-MC-NB) [17], blockchain-based privacy algorithm in a distributed database (DEPLSEST) [21] and multi-core DBSCAN [22]. Then the obtained results are illustrated in Table 2.

Table 2

CAI Analysis

Methods	Covertype	Mnist	SDDD
DISTPAB [16]	0.829	0.8124	0.765
K-MC-NB [17]	0.8024	0.7986	0.782
DEPLSEST [21]	0.849	0.815	0.80
MC-DBSCAN [22]	0.858	0.824	0.821
PPDC-ODL	0.890	0.924	0.934

Table 2 illustrates the CAI analysis of introduced privacy-preserving clustering with an optimized deep learning algorithm based on distributed dataset analysis. From the analysis, the introduced PPDC-ODL approach attains high CAI values on various three datasets such as Covertype, Mnist and SDDD. The PPDC-ODL approach has results from 0.89 to 0.934, which is higher than the other clustering algorithms. Here, the introduced PPDC-ODL approach uses the various numbers of hidden layers with hyperbolic tangent activation functions. The learning and activation function is more helpful in recognizing the training patterns with minimum computation complexity. Confidentiality controls help to guarantee that individuals and systems are adhering to their privacy obligations by ensuring that information is protected in accordance with privacy rules. For every iteration, the network works effectively and predicts the objective function to minimize the loss values of output. Then the respective graphical analysis is illustrated in Fig. 2.

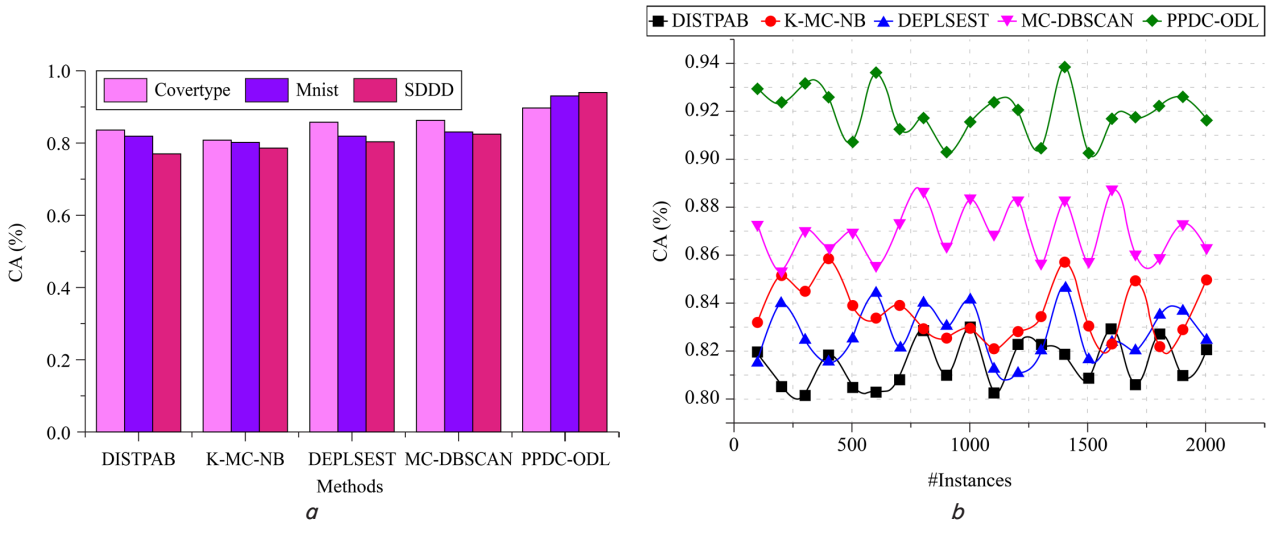

Fig. 2. Consistent accuracy index analysis: *a* — datasets; *b* — instances

Fig. 2 illustrates the CAI analysis of the introduced PPDC-ODL approach-based clustering process in the distributed datasets. Here, the collected datasets are investigated in respective hidden layers using network parameters such as weight and bias. The computed outputs in the hidden layers are concatenated:

$$J\left(\text{concat}\left(H_{jk}\right),H'_{jk}\right)=\text{concat}\left(H_{jk}\right)-H'^{2}_{jk}.$$

To obtain the global code or pattern. These concatenate results are more useful for investigating the global code and pattern with minimum computation complexity. The network uses the stochastic gradient descent learning procedure to update and optimize network performance. In addition, the network parameters are selected according to the genetic bee fitness function $fit_i \Big/ \sum_{j=1}^{PS} fit_j$, which reduces the deviation between the actual and predicted value. The minimum deviation clearly states the low difference between the centroid cluster output and distributed clustering outputs $m^c(c_{ij})$ and $m^D(c_{ij})$. Therefore, the introduced PPDC-ODL approach has a high CAI value compared to the other methods compared to the existing method.

5. 2. Accuracy analysis

In addition, the effectiveness of the secured clustering process efficiency is determined using the unsupervised clustering accuracy (ACC) evaluated on three datasets and the different number of instances. The ACC metrics help to understand how effectively the matching results are clustered between the actual and predicted values. The obtained ACC values are illustrated in Table 3.

Table 3 illustrates the ACC analysis of introduced privacy-preserving clustering with an optimized deep learning algorithm based on distributed dataset analysis. From the analysis, the introduced PPDC-ODL approach attains high ACC values on various three datasets such as Covertype, Mnist and SDDD. The PPDC-ODL approach has resulted from 0.922 to 0.967, higher than the other clustering algorithms. The optimized deep learning model uses the 10 hidden neurons in seven hidden layers. Each hidden layer has a specific activation function and adaptive learning procedure, which generates the replica reconstruction output. The predicted output values are more related to the local code that represents the global code. More ever, the generated outputs similarly match the centralized dataset local code. Then the graphical analysis of the results is illustrated in Fig. 3.

Table 3

ACC Analysis

Methods	Covertype	Mnist	SDDD
DISTPAB [16]]	0.878	0.893	0.879
K-MC-NB [17]	0.869	0.879	0.893
DEPLSEST [21]	0.892	0.883	0.892
MC-DBSCAN [22]	0.882	0.889	0.902
PPDC-ODL	0.922	0.930	0.967

Fig. 3 illustrates the ACC values of different clustering algorithms. From the analysis, the introduced PPDC-ODL approach attains a high unsupervised clustering accuracy (ACC) value compared to the existing methods. The effectiveness is determined with three datasets and respective instances. The neural network uses the various local dataset attributes to produce global patterns with minimum objective function loss $X_i - X_i'^2$. The computed replica reconstruction output from hidden layers is concatenated $\text{concat}\left(H_{jk}\right)-H'^{2}_{jk}$, minimizing the redundant data participation in clustering. This process minimizes the computation complexity while forming a similar cluster. The similar closest clusters are computed $[\gamma_{(i,1)} \cdot p_{(i,r)} \ldots . \gamma_{(i,K)} \cdot p_{(i,r)}]_H$ by examining each data in the group Gm. The effective computation of cluster centroid and distance between the data improved the overall clustering process with minimum deviation values.

5. 3. Error rate analysis

The system's effectiveness is further evaluated using the deviation or mean square error rate on various datasets and instances. Then the obtained results are illustrated in Fig. 4.

Fig. 4 illustrates the error rate analysis of the introduced PPDC-ODL clustering algorithm in distributed data mining (DDM). Compared with existing techniques, the introduced method attains the minimum deviation error. The low error value indicates that the system has similar local or global codes while clustering and classifying the replication data. The clustering process has a key generation process shared between the service provider. The service provider generates the database to maintain the distributed data security.

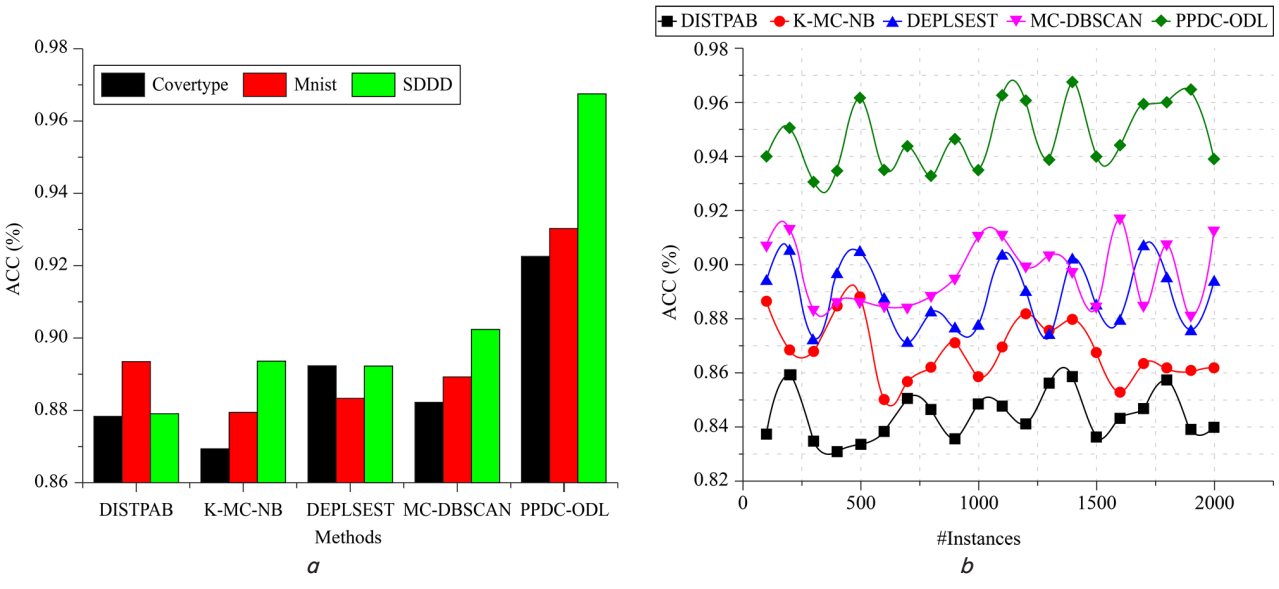

Fig. 3. Accuracy analysis of: *a* — datasets; *b* — instances

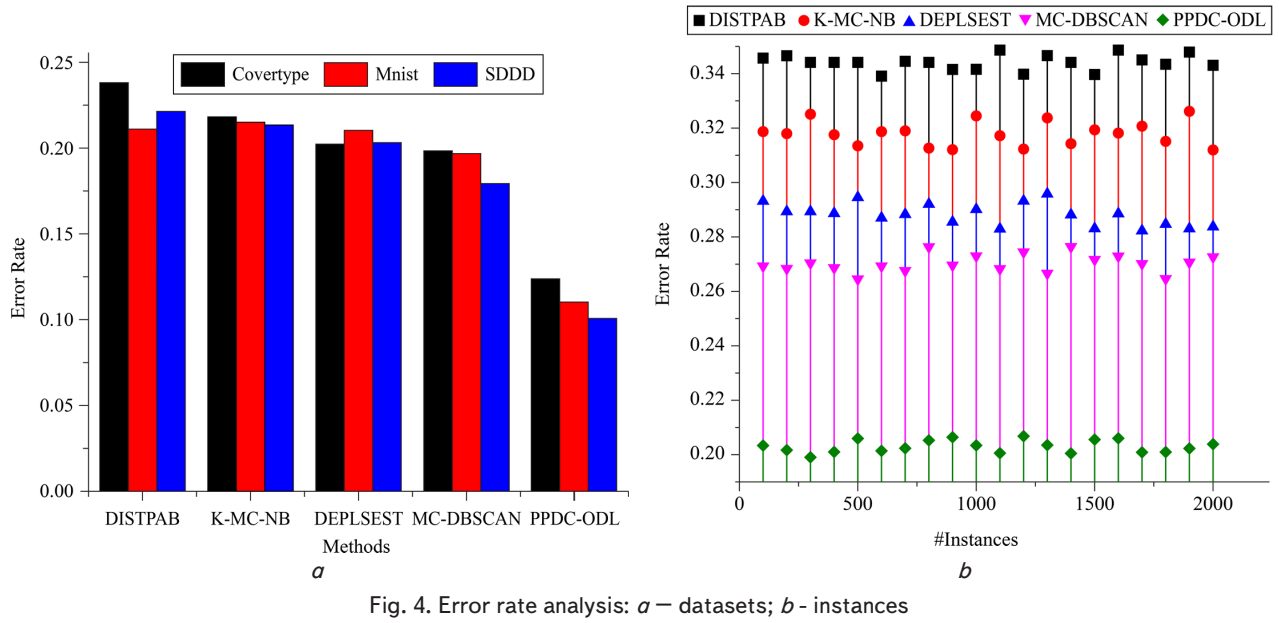

Fig. 4. Error rate analysis: *a* — datasets; *b* - instances

During the analysis, the system has a minimum error rate because, for every iteration, the closest data points are selected according to the distance measure:

$$\left[\left[D_i^2\right]\right]_H = \left[\left[C^2\right]\right]_H \cdot \prod_{r=1}^{R} \left[\left[C_r\right]\right]_H^{-2p_{(i,r)}} \cdot \left[\left[P^2\right]\right]_H.$$

The minimum distance value is chosen as the closest data point. Therefore, the deviation between the actual and predicted clustering values is very low compared to other methods. The minimum error rate is achieved by updating the network parameter with the respective optimization algorithm. The low error value means the system has fewer computational and optimization problems. Then the overall results for different distances are illustrated in Table 4.

Thus the system maintains data security while analyzing data during the clustering process. Here, the local data clustering eliminates the irrelevant and replicated data by analyzing data with the multiple hidden layer-based deep learning algorithm. For every time, clustering uses the encryption keys to manage the data security, effectively improving the further privacy transaction in the DDM environment.

Table 4

Overall Efficiency Analysis

Metrics	DIST-PAB	K-MC-NB	DEPLS-EST	MC-DB-SCAN	PPDC-ODL
CAI	0.816	0.839	0.826	0.868	0.920
ACC	0.842	0.868	0.890	0.899	0.950
Error Rate	0.344	0.318	0.289	0.272	0.202

Fig. 5 shows the energy costs of the proposed PPDC-ODL model. In modern energy security studies, the four A's (availability, affordability, accessibility, and acceptability) are often used as a starting point for the analysis. The devices investigate the data in petabytes and terabytes with thousands of processors. This device-based computation is

only applicable for huge volumes of data unsuitable for traditional systems.

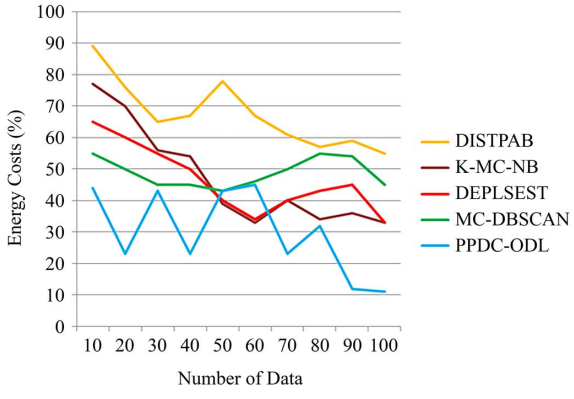

Fig. 5. Energy costs

6. Discussion of experimental results of privacy protection based distributed clustering (PPDC)

The excellence of data clustering process evaluated the optimized deep learning approach based. The method uses privacy protection-based clustering to ensure data security and privacy factors. The clustering algorithm investigates the relationship between the data in DDM. The correlation between the data is grouped. The effective computation of relationship identification helps to minimize the dimensionality issues. In addition, the gathered information was analyzed using the optimized deep learning approach. The neural model uses the hidden layer activation process with network parameters. Effective network parameter utilization minimizes the deviation between the actual and computed data. The low error rate value indicates that the system ensures high clustering data. In addition, the deep learning model uses the validation process to maintain data privacy while clustering. In addition, the network uses the training process that investigates the global code and pattern with minimum computation complexity. The obtained results are clearly illustrated in Table 2.

The introduced approach uses the privacy-preserving-based clustering process that helps to manage the data security while clustering. In addition, the clustering process investigates the data availability, trusted parties, semi-trusted third party information etc. These features help to manage data security and privacy. In addition, the optimized neural models are also incorporated to improve the overall data analysis and minimize the high-dimensional data. Even though the introduced method works effectively, the user should be validated strongly while accessing the data in the third-party server. The validation process helps maintain the authorization and authentication and eliminates the intermediate access in the DDM. Then the effectiveness of this objective is evaluated using the CAI metric, and the system ensures 0.92 % which is clearly described in Fig. 2. The Fig. 2, *a*, *b* investigates the CAI anlaysis for different methods and instances. From the analysis, the authorized evaluation achieved to manage the access controls while data clustering. Then the deviation between the actual and computed values is less (0.202) which is depicted in Fig. 4. The introduced method evaluated with different methods and instances and the system ensures the minimum deviation value. The minimum deviation value clearly illustrated that the method have minimum difficulties while investigating the high-dimensionality data. Even though the introduced method successfully identify and verify the user details while handling the frequent user information; the intermediator may access the information which leads to create the security issues. Therefore, the frequent user credential information should be more secured to reducing the risk factors in future data analysis.

7. Conclusions

1. The research uses the Privacy Preserving clustering approach that examines the user preference by validating the user's personal information. The user personal information validation helps to manage the data security and privacy. In addition, group members also investigated according to the data relationship, which minimizes the intermediate data involvement. Therefore, the privacy-preserving clustering process achieves 95 % accuracy in the data compared to other methods.

2. In this work, the optimized neural model is introduced that analyses the input data according to the encoder-decoder variable, which successfully minimizes high-dimensional computation issues. In addition, the data has been grouped by examining the data relationship computation. The effective examination of data links helps to reduce unwanted data participation. Then the effectiveness of this objective is evaluated using the CAI metric, and the system ensures 0.92 %.

3. The deep neural model uses the optimization technique to update the network parameter, which helps to minimize the deviation between the clustering process. In addition, the neural model selects the most relevant information from the various resources that resolve the high-dimensionality issue. Then the deviation between the actual and computed values is less (0.202).

These methods ensure the high privacy and confidentiality rate while accessing the data in the distributed network. However, the security and privacy should be improved while transferring the high sensitivity data.

Conflict of interest

The authors declare that they have no conflict of interest in relation to this research, whether financial, personal, authorship or otherwise, that could affect the research and its results presented in the paper.

Acknowledgments

The authors would like to thank to Technical Instructors Training Institute, Middle Technical University for supported, encourage and providing infrastructure to carry our research work.

References

1. Omidipoor, M., Toomanian, A., Neysani Samany, N., Mansourian, A. (2020). Knowledge Discovery Web Service for Spatial Data Infrastructures. ISPRS International Journal of Geo-Information, 10 (1), 12. doi: https://doi.org/10.3390/ijgi10010012

2. Liu, X., Huang, Q., Gao, S., Xia, J. (2021). Activity knowledge discovery: Detecting collective and individual activities with digital footprints and open source geographic data. Computers, Environment and Urban Systems, 85, 101551. doi: https://doi.org/10.1016/j.compenvurbsys.2020.101551
3. Qasem, M. H., Obeid, N., Hudaib, A., Almaiah, M. A., Al-Zahrani, A., Al-Khasawneh, A. (2021). Multi-Agent System Combined With Distributed Data Mining for Mutual Collaboration Classification. IEEE Access, 9, 70531–70547. doi: https://doi.org/10.1109/access.2021.3074125
4. Mewada, S. (2021). Data Mining-Based Privacy Preservation Technique for Medical Dataset Over Horizontal Partitioned. International Journal of E-Health and Medical Communications, 12 (5), 50–66. doi: https://doi.org/10.4018/ijehmc.20210901.oa4
5. Zhan, Z.-H., Shi, L., Tan, K. C., Zhang, J. (2021). A survey on evolutionary computation for complex continuous optimization. Artificial Intelligence Review, 55 (1), 59–110. doi: https://doi.org/10.1007/s10462-021-10042-y
6. Lee, J.-S., Jun, S.-P. (2021). Privacy-preserving data mining for open government data from heterogeneous sources. Government Information Quarterly, 38 (1), 101544. doi: https://doi.org/10.1016/j.giq.2020.101544
7. Cunha, M., Mendes, R., Vilela, J. P. (2021). A survey of privacy-preserving mechanisms for heterogeneous data types. Computer Science Review, 41, 100403. doi: https://doi.org/10.1016/j.cosrev.2021.100403
8. Du, G., Zhang, J., Li, S., Li, C. (2021). Learning from class-imbalance and heterogeneous data for 30-day hospital readmission. Neurocomputing, 420, 27–35. doi: https://doi.org/10.1016/j.neucom.2020.08.064
9. Soomro, T. A., Zheng, L., Afifi, A. J., Ali, A., Yin, M., Gao, J. (2021). Artificial intelligence (AI) for medical imaging to combat coronavirus disease (COVID-19): a detailed review with direction for future research. Artificial Intelligence Review, 55 (2), 1409–1439. doi: https://doi.org/10.1007/s10462-021-09985-z
10. Alomari, E., Katib, I., Albeshri, A., Yigitcanlar, T., Mehmood, R. (2021). Iktishaf+: A Big Data Tool with Automatic Labeling for Road Traffic Social Sensing and Event Detection Using Distributed Machine Learning. Sensors, 21 (9), 2993. doi: https://doi.org/10.3390/s21092993
11. Guo, Y., Zhao, R., Lai, S., Fan, L., Lei, X., Karagiannidis, G. K. (2022). Distributed Machine Learning for Multiuser Mobile Edge Computing Systems. IEEE Journal of Selected Topics in Signal Processing, 16 (3), 460–473. doi: https://doi.org/10.1109/jstsp.2022.3140660
12. Matsumoto, N., Hamakawa, Y., Tatsumura, K., Kudo, K. (2022). Distance-based clustering using QUBO formulations. Scientific Reports, 12 (1). doi: https://doi.org/10.1038/s41598-022-06559-z
13. Sharma, K. K., Seal, A. (2021). Spectral embedded generalized mean based k-nearest neighbors clustering with S-distance. Expert Systems with Applications, 169, 114326. doi: https://doi.org/10.1016/j.eswa.2020.114326
14. Kotsiopoulos, T., Sarigiannidis, P., Ioannidis, D., Tzovaras, D. (2021). Machine Learning and Deep Learning in smart manufacturing: The Smart Grid paradigm. Computer Science Review, 40, 100341. doi: https://doi.org/10.1016/j.cosrev.2020.100341
15. Du, J., Jiang, C., Gelenbe, E., Xu, L., Li, J., Ren, Y. (2018). Distributed Data Privacy Preservation in IoT Applications. IEEE Wireless Communications, 25 (6), 68–76. doi: https://doi.org/10.1109/mwc.2017.1800094
16. Chamikara, M. A. P., Bertok, P., Khalil, I., Liu, D., Camtepe, S. (2021). Privacy preserving distributed machine learning with federated learning. Computer Communications, 171, 112–125. doi: https://doi.org/10.1016/j.comcom.2021.02.014
17. Javid, T., Gupta, M. K., Gupta, A. (2022). A hybrid-security model for privacy-enhanced distributed data mining. Journal of King Saud University - Computer and Information Sciences, 34 (6), 3602–3614. doi: https://doi.org/10.1016/j.jksuci.2020.06.010
18. Xia, C., Hua, J., Tong, W., Zhong, S. (2020). Distributed *K*-Means clustering guaranteeing local differential privacy. Computers & Security, 90, 101699. doi: https://doi.org/10.1016/j.cose.2019.101699
19. Shewale, A., Keshavamurthy, B. N., Modi, C. N. (2018). An Efficient Approach for Privacy Preserving Distributed K-Means Clustering in Unsecured Environment. Recent Findings in Intelligent Computing Techniques, 425–431. doi: https://doi.org/10.1007/978-981-10-8639-7_44
20. Xiong, J., Ren, J., Chen, L., Yao, Z., Lin, M., Wu, D., Niu, B. (2019). Enhancing Privacy and Availability for Data Clustering in Intelligent Electrical Service of IoT. IEEE Internet of Things Journal, 6 (2), 1530–1540. doi: https://doi.org/10.1109/jiot.2018.2842773
21. Chen, Y., Xie, H., Lv, K., Wei, S., Hu, C. (2019). DEPLEST: A blockchain-based privacy-preserving distributed database toward user behaviors in social networks. Information Sciences, 501, 100–117. doi: https://doi.org/10.1016/j.ins.2019.05.092
22. Ni, L., Li, C., Wang, X., Jiang, H., Yu, J. (2018). DP-MCDBSCAN: Differential Privacy Preserving Multi-Core DBSCAN Clustering for Network User Data. IEEE Access, 6, 21053–21063. doi: https://doi.org/10.1109/access.2018.2824798
23. Zhang, T., Zhu, Q. (2018). Distributed Privacy-Preserving Collaborative Intrusion Detection Systems for VANETs. IEEE Transactions on Signal and Information Processing over Networks, 4 (1), 148–161. doi: https://doi.org/10.1109/tsipn.2018.2801622
24. MNIST Database of Handwritten Digits. URL: https://archive-beta.ics.uci.edu/ml/datasets/mnist+database+of+handwritten+digits
25. Covertype Data Set. URL: https://archive.ics.uci.edu/ml/datasets/covertype
26. Dataset for Sensorless Drive Diagnosis Data Set. URL: https://archive.ics.uci.edu/ml/datasets/dataset+for+sensorless+drive+diagnosis