Article

Not peer-reviewed version

Big Data Privacy Protection and Security Provisions of Healthcare SecPri-BGMPOP Method in Cloud Environment

Moorthi K , Jothi Prabha Appadurai , Balasubramanian Prabhu Kavin , Jeeva Selvaraj , Hong-Seng Gan , Wen-Cheng Lai *

Posted Date: 4 April 2024

doi: 10.20944/preprints202404.0386.v1

Keywords: Big Data; Security; Privacy; Boost Graph convolutional network clustering algorithm; Magnify Pinpointing based encryption approach; Hybrid Particle swarm; Grey wolf optimization

Article

Big Data Privacy Protection and Security Provisions of Healthcare SecPri-BGMPOP Method in Cloud Environment

Moorthi.K [1], Jothi Prabha Appadurai [2], Balasubramanian Prabhu kavin [3], Jeeva Selvaraj [4], Gan Hong Seng [5] and Wen-Cheng Lai [6,*]

[1] Department of Computational Intelligence, College of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur - 603203, Chengalpattu Dist. Tamilnadu, India.; moorthicse@gmail.com
[2] Department of CSE(Networks); Kakatiya institute of technology and science, Warangal - 506015, Telangana. ajp.csn@kitsw.ac.in
[3] Department of Data Science and Business Systems, College of Engineering and Technology, SRM Institute of Science and Technology, SRM Nagar, Kattankulathur - 603203, Chengalpattu Dist. Tamilnadu, India; ceaserkavin@gmail.com
[4] Department of Information Science and Engineering, Jain Deemed to be university, Global campus, Bangalore, India. jeeva.selvaraj@jainuniversity.ac.in;
[5] School of AI and Advanced Computing, XJTLU Entrepreneur College (Taicang), Xi'an Jiaotong - Liverpool University, Suzhou, Jiangsu, P.R. China 215400. hongSeng.Gan@xjtlu.edu.cn
[6] Department of Electrical Engineering, Ming Chi University of Technology, Taiwan
* Correspondence: wenlai@mail.mcut.edu.tw; wenlai@mail.ntust.edu.tw

Abstract: One of the industries with the fastest rate of growth is healthcare, and this industry's enormous data requires extensive cloud storage. The cloud may offer some protection, but there is no assurance that data owners can rely on it for refuge and privacy amenities. Therefore, it is essential to offer security and privacy protection. However, maintaining privacy and security in an untrusted green cloud environment is difficult, thus the data owner should have complete data control. A new work SecPri-BGMPOP(security and privacy of Boost graph Convolutional Network -Pinpointing-Optimization Performance) is suggested that can offer a solution that involves several different steps in order to handle the numerous problems relating to security and protecting privacy. Initially Boost Graph Convolutional Network Clustering (BGCNC) algorithm, which reduces computational complexity in terms of time and memory measurements, is first applied to the input dataset to begin the clustering process. Second, enlarge by employing a piece of the magnifying bit string to generate a safe key, pinpointing based encryption avoids amplify leakage even if a rival or attacker decrypts the key or asymmetric encryption. Finally, to determine the accuracy of the method, an optimal key was created using a meta-heuristic algorithmic framework called Hybrid fragment horde bland lobo Optimization (HFHBLO). Our proposed method currently kept in a cloud environment, allowing analytics users to utilize it without risking their privacy or security.

Keywords: big data; security; privacy; boost graph convolutional network clustering algorithm; magnify pinpointing based encryption approach; hybrid particle swarm; grey wolf optimization

1. Introduction

There are now many electronic medical records, hospital information systems, medical imaging, and other types of data due to the Internet's quick expansion and the analysis of big data, which has gradually saturated the medical business. According to experts in linked sectors, the amount of data in the medical industry will be 44 times larger by 2020 than it was in 2009 [1]. The medical IoT network gathers physiological data, which is then sent to the healthcare big data centre for archival and disease diagnostics. The medical file must be encrypted before transmission in order to preserve patients' privacy and stop third parties from listening in on private communications. On the protected data, the patient applies an access policy to define the permissible attributes and relationships. Only users possessing the appropriate attribute secret keys—such as a doctor, nurse, anaesthetist, or a patient's family—are allowed to decrypt the ciphertext. Attribute-based encryption is the name of this encryption technique [2]. Information security and protection has evolved into a fundamental problem that impacts numerous cloud applications. The ease with which cloud administrators can access sensitive data is one of the major concerns with regard to data security and privacy. This concern sharply increases client anxiety and hinders the adoption of distributed computing in many industries, including the financial sector and governmental and administrative entities. The security of structured or unstructured data is not addressed by the conventional Big Data approach. In cases where the risk of disclosing personal information is real, it also has to combine the concepts of insurance and security. Because of the enormous combination of organized and big data necessitates new models for improving safety and security. The growing issue of data security in big data is of great interest to researchers [3]. A patient with unclear symptoms may be diagnosed and treated in many hospitals, and various medical records may be kept, according to the current medical system. As a result, developing a cross-domain safe data sharing system is essential to facilitating patient care at various hospitals. For instance, the clinicians at hospital B have access to the examination report created by hospital A. The public clouds are used to store and provide ubiquitous data access for the encrypted medical files produced by various hospitals [4], [5]. The cross-domain access policy for a patient's protected medical records is determined by the patient. Each member of the medical staff must register with their respective medical facility in order to obtain the attribute secret key needed to decrypt the patient's encrypted files[6,7]. The below Figure 1 explains the role of Big Data in Healthcare Industry. The sources of big data used in healthcare range from wearable technology and search engine server logs to electronic health records. This is the limitless ocean of information that presents an infinite number of opportunities. Knowing how to use this data effectively is crucial. All parties involved in the healthcare system, including as healthcare organisations (HCOs), patients, medical professionals, pharmaceutical makers, etc., can benefit from proper storage and analysis instruments. In general, patients' health improves, doctors can dramatically improve medical results, HCOs can reduce costs and increase operational efficiency, pharmaceutical companies can make better judgements, and other healthcare providers can do so as well.

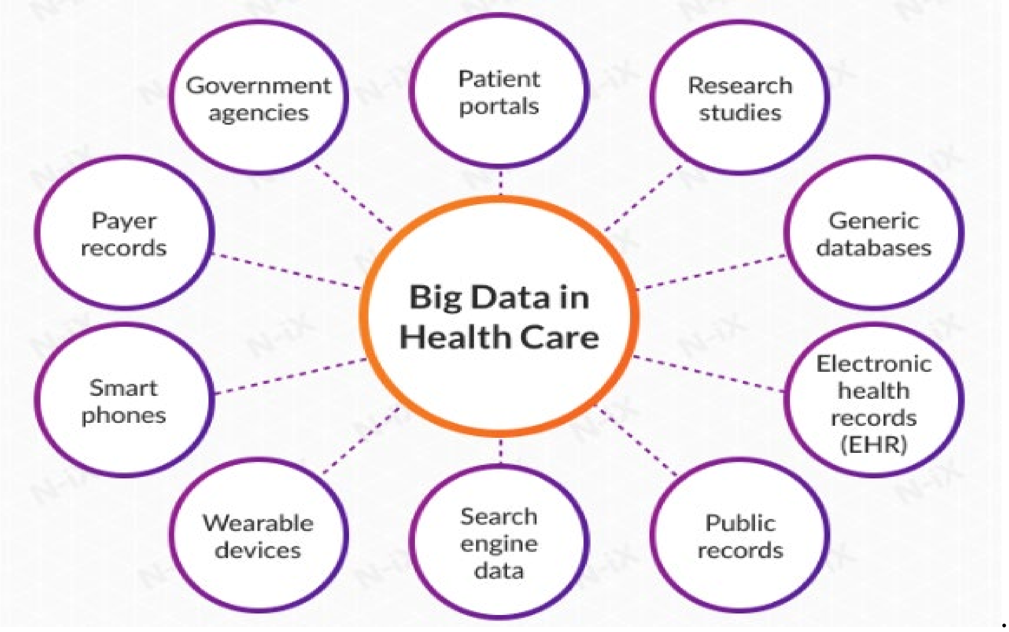


Figure 1. Big Data in Healthcare Industry.

Unencrypted data transmissions are one of the privacy issues brought on by the volume of data [8,9]. The Cloud server, as well as any Database server in the Cloud, is unreliable. The Febo is a *comprehensive health toolbox* which is helps you keep track of your bowel motions and discover how your body reacts. [10]. Green computing that uses less energy includes [11], which is used to create the best plans for assigning tasks, and [12,13], which addresses energy waste issues in dynamic networking environments. The goal of both of these examples is to cut down on overall energy use.

One of our primary areas of focus will be healthcare. Because each person's genome occupies more than 140 terabytes, healthcare applications require a significant to safely store genome data [14,15]. As a result of security breaches [16–18], data owners reveal on big data servers or in the Cloud. Password guessing, brute-force attacks, stolen verification attacks, and other security threats target big data storage systems. Users' and data owners' confidentiality is not adequately protected by current security measures, which advocate encrypting data before delivering it [19,20].

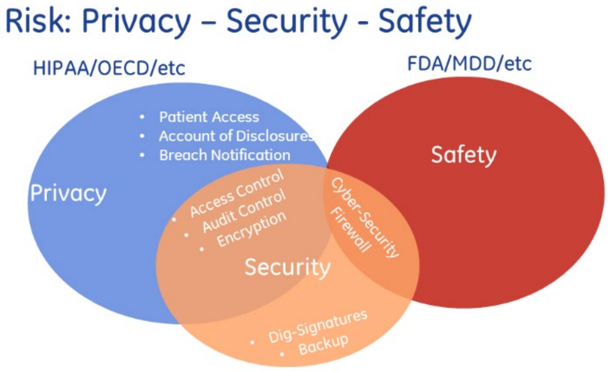


Figure 2. Big Data Privacy Assurance in Healthcare System.

As the above Figure 2 clearly portrays the privacy assurance of Big Data in Healthcare system, a SecPri-BGMPOP, which can offer a solution that incorporates multiple processes, is proposed as a means of addressing the numerous issues associated with privacy and security protection. A brand-new Boost Graph convolutional network clustering (BGCNC) algorithm was initially used to develop on effective techniques. Provide a framework to improve BGCNC's convergence to advance this idea. The advantages we provide in terms of memory and computation are significant. It has the same memory complexity as conventional SGD and full gradient descent have the same temporal complexity per epoch. Second, a Magnify Pinpointing-based encryption(MPBE) method that uses a portion of the identification bit string to create a safe key, preventing identity leakage even if the encryption or encoded data is cracked by an enemy or assault. Using the meta-heuristic algorithmic framework Hybrid fragment horde bland lobo Optimization (HPSGWO), an optimal key was then produced. Despite their good performance, traditional algorithms can be improved upon to address their flaws and raise the bar. The traditional PSO method has some drawbacks, such as subpar performance in a number of sectors. The GWO algorithm also has slower convergence, less solution precision, and less efficient local searching in addition to these drawbacks. The updated place is the primary renovation in the proposed paradigm. Our recommended approach is now maintained in a cloud environment, enabling analytics users to access it without jeopardising their security or privacy.

2. Literature Review

One of the most prominent research fields at the moment is cloud security, and several methods have been put out over the previous few decades. Big data analytics are becoming increasingly significant for SDN-based smart grids. A possible strategy is to examine the vast amounts of data produced by an SDN-based smart grid by utilizing machine learning techniques. ODPCA, a technique for selectively private and optimum clustering. For privacy-preserving clustering of mixed data, the ODPCA combines the differentially private K-means algorithm with the K-modes approach. The distribution of the privacy budget is optimized [21] to boost the accuracy of the clustering findings. Using fog computing technology, secure healthcare sensitive data may be kept in the cloud. A tri-party one-round authenticated key-agreement mechanism that may generate a session key among participants and enable safe communication can also be represented by bilinear pairing cryptography. Finally, a decoy method might be used to access and safeguard private healthcare data. The suggested method [22] states that when an intruder is discovered accessing system data, fake files are retrieved right away to protect data security.

A Blockchain-based system that uses secure key management (BC-EKM). In order to build the stake blockchain, a hybrid sensor network is used. For key management, safe cluster formation and node movement algorithms are also built, with the stake blockchain serving as a trust machine in place of most of the BS's duties. Finally, we carry out in-depth security simulations and analyses [23]. For a number of smart city applications, Holistic Big Data Integrated Artificial Intelligent Modelling (HBDIAIM) has been presented as a way to enhance the security and privacy of the data management interface. To adequately secure the private data management interface in smart city applications, a differential evolutionary algorithm has been incorporated to HBDIAIM. The differential evolutionary algorithm also incorporates the Big Data analytics assisted decision privacy approach, which enhances the scalability and accessibility of data in a data management interface based on their corresponding storage location. Additionally, to solve privacy and scalability issues in the data management interface for various smart city applications, the Adaptable Interference Method was created [24].

By recommending a better key management system, this study aims to address the challenges related to the security and privacy concerns of sensitive patient information. The suggested approach also strives to offer a simple and structured key management system. This system has a null rekeying technique and only needs a few key calculations to achieve forward and backward compatibility secrecy. As a consequence, the Healthcare Key Management (HCKM) framework, which strives to decode the same plain text with various keys, is the safe and privacy-preserving key management method for e-health systems. While maintaining an adequate degree of security, HCKM reduces the rekeying overhead for group members and the overhead stated in terms of the number of messages exchanged [25]. a blockchain-based Internet of Things solution for large data transport and privacy protection facilitated by AI. Using graph modeling, the proposed algorithm first constructs a reliable and scalable system for data collection and transmission. Additionally, the approach based on artificial intelligence is used to extract the subset of nodes and produce efficient healthcare services. Using symmetric-based digital certificates, blockchain allows for secure and private transmission of communication resources [26].

By lengthening the keys in Data Encryption Standard, the TDES technique offers a way that is significantly easier to use while yet securing data privacy and preventing assaults (DES). The experiment's findings demonstrate that the TDES technique is effective in securing and safeguarding substantial volumes of healthcare data stored in the Cloud[27]. This research presents a unique encryption technique using Serpent, Advanced Encryption Standard (AES), and elliptic curve cryptography to protect healthcare data in IoT-enabled healthcare infrastructure. The suggested hybrid encryption approach improves healthcare data security by integrating symmetric and asymmetric-based encryption techniques. This technique additionally makes use of an elliptic curve-based digital signature to ensure the data's integrity [28]. The IoMT network receives massive amounts of big data, also known as health data, and registers a sizable number of patients and devices each day. To prevent misuse, this patient data should remain private and secure on the IoMT network. To achieve such data privacy and security, the interplanetary file system (IPFS) and a three-level/tier network have been proposed[29].

Enhanced architecture for safe and scalable IoT-based healthcare data transfer, based on the routing protocol. First, a variety of Internet of Things (IoT) devices, including wearables and sensors, collect health data. Utilizing data reduction and cleaning procedures, the raw data are preprocessed. Principal Component Analysis (PCA) and K-Nearest Neighbour (KNN) imputation are used to minimize the dimensionality of the data. The preprocessed data is used to extract the features using modified local binary patterns (MLBP). The FDT-RPL protocol for low-power and lossy networks enhances overall data transmission security by integrating the Butter Ant Optimization (BAO) algorithm with the fuzzy dynamic trust-based RPL algorithm [30]. Our recommended approach offers a more reliable security system for cloud-based large data in healthcare.

doi:10.20944/preprints202404.0386.v1

The remaining parts of the paper are laid out as follows: The proposed work is discussed in Section 3, Results and Discussions are presented in Section 4, Finally, the conclusion and the work to come are discussed in Section 5

3. Proposed Method

The SecPri-BGMPOP approach is suggested that can offer a solution that involves several different steps in order to handle the numerous problems relating to security and protecting privacy. Initially Boost Graph Convolutional Network Clustering (BGCNC) algorithm, which reduces computational complexity in terms of time and memory measurements, is first applied to the input dataset to begin the clustering process. The Boost Graph convolutional network clustering operates is as follows: BGCNC training algorithm that is fast and memory efficient. It samples a block of nodes linked to a dense subnetwork at each phase and restricts the neighbourhood search towards this substring. The dense subgraph is identified via a graph clustering algorithm. The memory and computational efficiency are greatly increased by using this straightforward yet efficient technique. An encryption method based on Magnify Pinpointing that prevents identity leaks even if an adversary or attacker decodes the key or encrypted material by creating a safe key utilising a portion of the identification bit string. Based on the Hybrid fragment horde bland lobo Optimization framework, an enhanced meta-heuristic algorithmic framework, an optimum key was obtained. Our suggested SecPri-BGMPOP is intended to offer a better security method for cloud-based big data in healthcare.

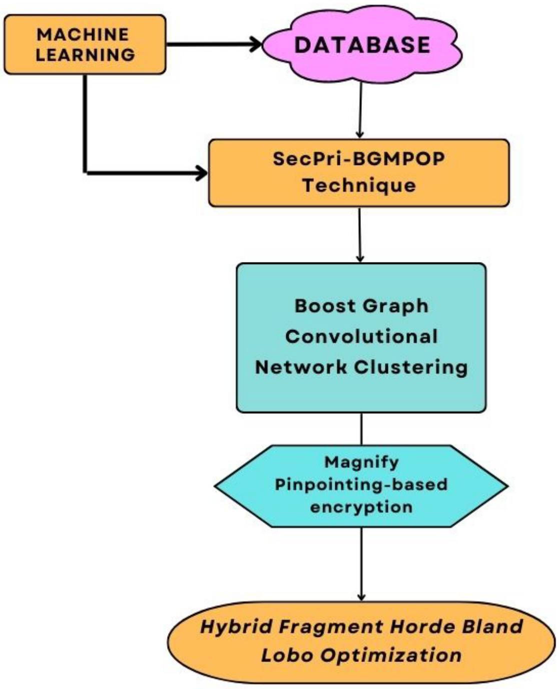

MACHINE LEARNING
DATABASE
SecPri-BGMPOP Technique
Boost Graph Convolutional Network Clustering
Magnify Pinpointing-based encryption
Hybrid Fragment Horde Bland Lobo Optimization

Figure 3. The main architecture of the data security and privacy model.

3.1. Graph Convolutional Network Clustering (GCNC)

The best of both worlds is achieved by Graph convolutional network clustering technique, this has the same memory complexity as conventional SGD and full gradient descent have the same temporal complexity per epoch. Consider the scenario in which each batch's embeddings are determined for a collection of nodes B from layer 1 to layer L. Given that every layer of computation uses the same subgraph $A_{B,B}(Links\ withi\ B)$ we may assume that the amount of edges in this batch, $\|A_{B,B}\|_0$. To indicates the embedding utilisation. we need build a batch B that maximises within-batch edges in order to increase embedding utilisation. In order to link the effectiveness of SGD updates with graph clustering techniques, Figure 1 shows the complete graph G and the graph with the clustering partition G as examples of the community expansion. Cluster-GCN, as can be shown, can concentrate on the neighbours within each cluster rather than conducting a thorough neighbourhood search.

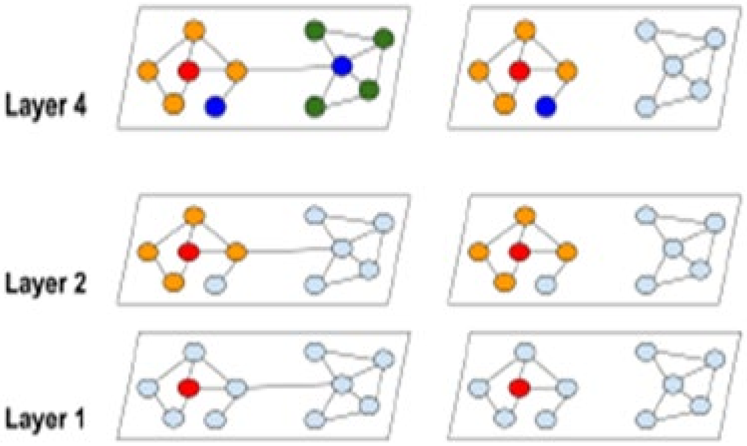


Figure 4. The distinction between our suggested cluster technique and conventional graph convolution is the neighbourhood expansion. The extension of neighbourhood nodes begins at the red node. Reflects the growing convolution suffers from hyperbolic neighbourhood growth, however our technique can stop expensive neighbourhood expansion.

For a graph $G = (V, \varepsilon, A)$ which consists of N=$|V|$ vertices and $|\varepsilon|$ edges such that an edge among any two vertices i and j represents their similarity. The corresponding adjacency matrix A is an $N \times N$ sparse matrix with (i, j) entry equaling to 1 if there is an edge between i and j and 0 otherwise. G's partition its nodes into c groups: $V = [V_1, \dots V_c]$, where V_t represents consists of the nodes in the t-th partition. L is number of layers, F is number of features, N is number of nodes, b is the batch size. As a result, we have c sub graphs

$$\bar{G} = [G_1, \dots G_c] = [\{v_1, \varepsilon_1\}, \dots, \{v_c, \varepsilon_c\}] \quad (1)$$

where each ε_t is made up only of the connections connecting the nodes in v_t. The adjacency matrix is divided into c^2 submatrices after nodes are reorganised as

$$A = \bar{A} + \Delta = \begin{bmatrix} A_{11} & 1 & A_{1c} \\ 1 & 0 & 1 \\ A_{c1} & 1 & A_{cc} \end{bmatrix} \quad (2)$$

And

$$\bar{A} = \begin{bmatrix} A_{11} & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & A_{cc} \end{bmatrix}, \Delta = \begin{bmatrix} 0 & 1 & A_{1c} \\ 1 & 0 & 1 \\ A_{c1} & 1 & 0 \end{bmatrix} \quad (3)$$

The links within G_t are contained in each diagonal block A_{tt} is a $|v_t| \times |v_t|$ adjacency matrix. The adjacency matrix for the graph G_t. $\bar{A}$ is the adjacency matrix for graph $\bar{G}$; A_{st} contains the connections between its two partitions v_t ;Δ . Similar to this, we can divide the $\hat{X}$ and Y into $[V_1, \dots V_c\ ]$ as $[X_1, \dots X_c\ ]$ according to the partition $[Y_1, \dots Y_c\ ]$ where X_t and $Y_t$are made up.

The calculation $\bar{G}$ is that it allows for the decomposition of the objective function of GCN into many groups (clusters). The final embedding matrix is if we use $\bar{A}'$ to represent the normalised version of $\bar{A}$.

$$Z^{(L)} = \bar{A}' \ \sigma\left(\bar{A}'\sigma\left(\dots\sigma\left(\bar{A}'XW^{(0)}\right)W^{(1)}\right)\dots\right)W^{(L-1)} \quad (4)$$

Significantly less difficult to construct than the neighbourhood search method utilised in earlier SGD-based training methods. Because $\bar{A}$ has a block-diagonal structure (remember that $\bar{A}'_{tt}$ is the matching diagonal block of $\overline{A}'$). Additionally, it has broken down into.

$$L_{\overline{A'}} = \sum_t \frac{|v_t|}{N} L_{\bar{A}'_{tt}} \text{ and } L_{\bar{A}'_{tt}} = \frac{1}{|v_t|} \sum_{i \in v_t} loss(yi, z_i^{(L)}) \quad (5)$$

The decomposition form in (3) and (4) serves as the basis for the Cluster-GCN (5). At each step, we sample a cluster V_t and the conduct SGD to update based on the gradient of $L_{\bar{A}'_{tt}}$ and this only requires the sub graph A_{tt}, the X_t, the Y_t on the current batch models $\{w^l\}_{l=1}^L$ Compared to the neighbourhood search strategy employed in earlier SGD-based training methods, the implementation merely calls for matrix components' forward and retrograde transmission (one block of (5)). As previously mentioned, the embedding utilisation is similar to the within-cluster linkages for each, therefore these are exactly what we require batch.1)Assuming that each node and its neighbours are often found in the same cluster, neighbourhood nodes have a high probability of remaining in the same cluster after a few hops. 2) We need to design a partition to reduce the amount of between-cluster links since we replace A with its block diagonal approximation A and the error is proportional to those links.

Each node in v_t of A_{tt} because each node only links to other nodes inside of v_t. Each batch's computation will only involve matrix products $\bar{A}'_{tt}X_t^{(l)}w^{(l)}$ and a few element-wise operations, thus the total computation time is $O(\|A_{tt}\|_0 F + bF^2)$. As a result, $O(\|A\|_0 F + NF^2)$ is the overall temporal complexity per epoch. On average, each batch only needs to compute $O(bL)$ embeddings, which is exponential to L rather than linear. For embedding storage, we simply must load b examples per batch and per layer's extracted features are saved using O(bLF) memory.

Boost Graph Convolutional Network Clustering (BGCNC)

To achieves good computational and memory complexity, two potential concerns still exist:

- Certain links (the part in Δ Eq. (2)) are eliminated from the graph after it has been partitioned. As a result, the performance can be impacted.
- Algorithms for graph clustering frequently group related nodes together. As a result, a cluster's distribution may deviate from the original data set, which could cause a skewed estimation of the complete gradient while doing SGD updates.

Based on the label distribution of each cluster, we determine its entropy value. When compared to random partitioning, it is evident that most clusters have lower entropies, which suggests that the label distributions of the clusters are skewed towards particular labels. This raises the variance between batches and could have an impact on SGD convergence.

We suggest a stochastic multiple clustering strategy to incorporate between-cluster linkages and lessen volatility across batches in order to overcome the aforementioned problems. Then, we divide the graph into p clusters $V_1 \dots \dots V_P$using a sizable p. Instead of taking into account just one cluster while building a batch B for an SGD update, we pick q clusters at random, denoted as $t_1, \dots \dots \dots t_q$ and add their nodes as part of the batch using the notation $\left\{V_{t_1} \cup \dots .. \cup V_{t_q}\right\}$. Moreover, the connections between the selected clusters,

$$\{A_{ij} | i, j \in t_1, \dots .. t_q\} \quad (6)$$

are re-added. In this way, the cluster combinations reduce the variance between batches and reintegrate the between-cluster linkages. Algorithm 1 presents our final Boost Graph convolutional network clustering technique.

```
Algorithm 1:Boost Graph convolutional network clustering
Input: Graph A, Feature X, Label Y;
Output: Node representation$\hat{X}$
Partition graph nodes into $c$ clusters $V_1, V_2, \dots \dots V_c$
$For\ iter = 1, \dots \max_{iter}\ do$
        Randomly choose $q$ clusters, $t_1 \dots . t_q$ from V without replacement;
        From the subgraph $\hat{G}$ with nodes $\hat{V} = \left[V_{t_1}, V_{t_2}, \ldots\ldots, V_{t_q}\right]$ and links $A_{\hat{V},\hat{V}}$;
        Compute $g \leftarrow \nabla L_{A_{\hat{V},\hat{V}}}$(loss on the subgraph $A_{\hat{V},\hat{V}}$);
        Conduct adam update using gradient estimator $g$
Output:$\{W_l\}_{l=1}^{L}$
```

The data is divided into groups using the Boost Graph convolutional network clustering technique, and each group is then handed to a faster parallel process to increase the computational complexity in terms of time and memory measurements. The Magnify Pinpointing-based encryption method is then applied to the data groups, which transforms the data into a different format based on the key value generated. To save time, the encryption, decryption, and key creation processes all use the Magnify Pinpointing-based approach.

3.2. Magnify Pinpointing-Based Encryption(MPBE)

Only key creation, encryption, and decryption times are calculated in the suggested method. Currently, network overhead is a problem. This framework shows how user 1 and user 2 can access each other's data by using magnify Pinpointing-based encryption as a security, authentication, and authorization tool. Both user 1 and user 2 two users who access the cloud's services, is shown in Figure 5 as part of the cloud environment using the cloud database for access.

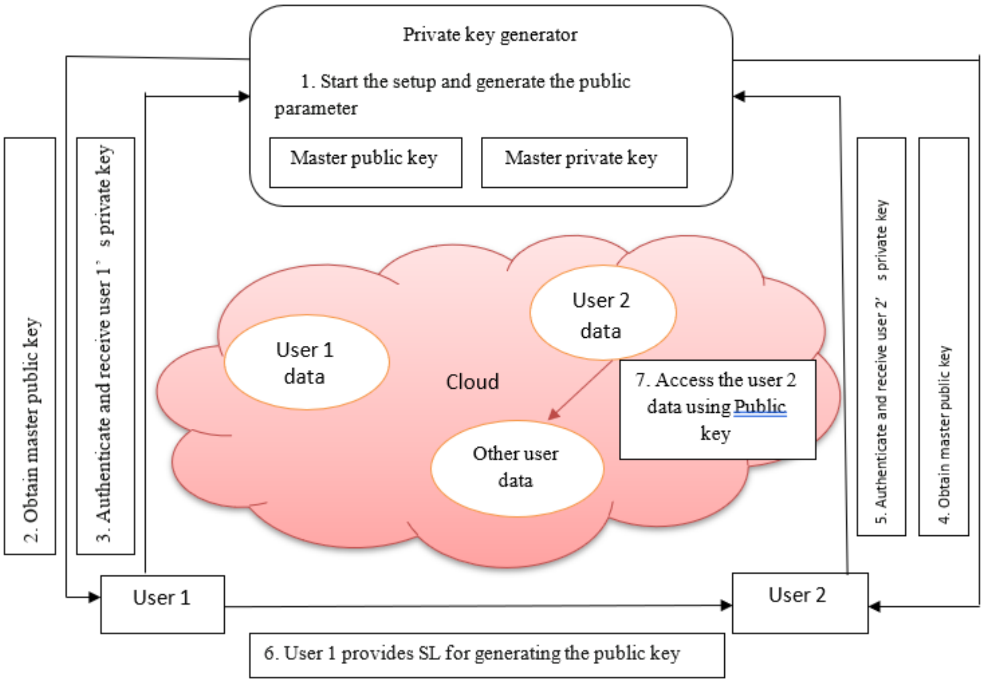

Private key generator
1. Start the setup and generate the public parameter
Master public key
Master private key
2. Obtain master public key
3. Authenticate and receive user 1' s private key
User 1 data
Cloud
User 2 data
7. Access the user 2 data using Public key
Other user data
5. Authenticate and receive user 2' s private key
4. Obtain master public key
User 1
User 2
6. User 1 provides SL for generating the public key

Figure 5. Framework for proposed work.

1. Private key generator (PKG) starts the setup process and decides the security parameters like the level of bits and type of curves.
2. Bob obtains the master public key from PKG.
3. Bob authenticates himself by issuing his identity to PKG and receives the private key for encrypting the data.
4. Similarly, Alice obtains the master public key from PKG.
5. Alice authenticates herself by issuing his identity to PKG and receives the private key for encrypting the data.
6. Bob sends his identity to Alice for generating the public key related to Bob's identity. Alice will use this key to decrypt the data
authenticated by Bob.
7. Alice gets the Bob's data from database and decrypts it for accessing.

Role-based classification is faster with identity-based encryption. With these techniques, users are given access to the data after authenticating their identities. This plan is initially put into practise in proxy servers to remove unwanted users. The identities of allowed users are recorded in proxy servers, and whenever an unauthorised user tries to access the server's service, their access is revoked because no key matches that identity. As a result, in order to access the service, each user must register their identification. Security is becoming increasingly more in demand as a result of technological advancements like cloud computing. This system is used by several security companies to protect user data and information. Uploading and downloading of data or files is a crucial component of cloud security implementation. The effectiveness of an IBE scheme is also significantly influenced by network congestion and service speed. However, in order to improve time efficiency, the suggested method described in Section 3.2 focuses primarily on key creation, encryption, and decryption. The only times that are calculated in the proposed method are those for key generation, encryption, and decryption. At this time, network overhead is compromised. Using IBE as a security, authentication, and authorization tool, this system illustrates how Bob and Alice can access each other's data An authorised user's identity cannot be obtained by an unauthorised user because PKG stores user identities in Lagrange polynomial equation form, which is computationally challenging for an outsider to decipher and obtain the user's original identity. Because they communicate these identities using secure socket connections (SSL), when Bob sends his identification to Alice in the middle of the network, an attacker cannot obtain the identity. Hash-based key generation techniques are computationally costly, hence pairing-based key generation techniques are used instead.

Security is becoming increasingly more in demand as a result of technological advancements like cloud computing. This system is used by several security companies to protect user data and information. Uploading and downloading of data or files is a crucial component of cloud security implementation.

Let G stand for a collection of prime order p. Group G forms an accurate bilinear map onto Group G_1. G_1's bilinear map representation is given by the formula e: $G \times G_1 \rightarrow G_2$, where g is the group G's generator. The group size is determined by a security parameter, and each identity is represented by four strings, each of length $n.4$.

$$SL = (sl_1, sl_2, sl_3, , , , sl_n). \tag{7}$$

From the SL bit strings length, utilised to produce fixed length n bit strings. The suggested Magnify Pinpointing-based encryption algorithm includes the following phases.

3.2.1. Initial Phase

Create the system parameters first. From Z_p a underground is chosen at random. Pick an accidental maker g from the set G such that $g \in G$, fix the value $gI = g^{\alpha}$, and randomly choose g_2 from the set G. Select a chance quantity u such that $u' \in G$ and a random n-length vector such that $U = \{u_i\}$ fuig after selecting all authority parameters. Finally, the public parameters $g, g1, g2, u'$, and U are broadcast along with the master key g_2^{α}.

3.2.2. Generation Phase

Let v be the n-bit string SL for the user, and let $V \subseteq \{1, \dots n\}$ be the collection of all i for which $v_i = 1$. V is split into two different, namely $V = \{v_1, v_2, \dots v_m\}$ and $\{V_{r_1}, V_{r_2}, \dots v_{r_m}\}$ such that $m + r_m = n$, where V_{r_1} stands for a random value that is introduced to the suggested approach to increase security. The private key corresponding to identity v is obtained by selecting a random value, as indicated in Equation (8).$u' \prod_{i \in V} v_i$ is the Formula for group operation during the key generation

$$d_v = \left(u' \prod_{i \in V} v_i \right) \tag{8}$$

$U = \{u_1, u_2 | \dots u_n\}$ and V=$\{v_1, v_2 \dots v_m\}$such that $m < n$. Now, create a polynomial function using the Lagrange coefficient method and perform polynomial interpolation. We can conceal of v that can be effectively reconstructed from the available data points with the aid of polynomial interpolation. For the suggested strategy, the polynomial equation and Lagrange coefficient is

Lagrange coefficient is

$$\Delta_{i,v}(x) = \sum_{i=0,k\in V}^{n} \left(\prod_{0<i<n,j\neq i} \frac{x - x_j}{x_i - x_j} \right) y_k \tag{9}$$

Where $x = u_i \, and \, y = v_k$.

Every user identity's random set u_i is generated once, and each identity's Lagrange coefficient is generated using the same u_i value. M-terms of identity will be used by authority value as a result of which a challenger will never learn the authentic user's original identity. As a result, it will be challenging to retrieve or deduce the key created for a specific SL. What if every user identity and U value is identical? The challenger will be unable to infer anything from the key in such case since $\Delta_{i,v}(x)$ produces an error of zero. This situation is unique. The greatest error between any two subsequent nodes will be shown, and the error created will be zero.

3.2.3. Encryption Phase

Let message $M(M \in G_1)$ and "c" be a random number selected fromZ_p. Equation can be used for encryption for some identities v. (10). Encryption using three-key TDEA is prohibited unless otherwise approved by another NIST guidance.

$$= \left(e\,(g_1, g_2)^c M, g^c, \left(u' \prod_{i=V} V_i \right)^c \right) \tag{10}$$

3.2.4. Decryption Phase

Let $C = (C_1, C_2, C_3)$ be a valid cypher text under user identity v for message M. Then, using cypher text C as a key, $d_v = (d_1, \, d_2)$ as given in Equation (11)-(13).

$$C_1 \frac{e(d_2, C_3)}{e(d_1, C_2)} = (e\,(g_1, g_2)^c M) \frac{e(g^r, (u' \prod_{i=v} v_i)^c)}{e(g_2^\alpha, (u, \prod_{i=v} v_i)^r, g^c)}, \tag{11}$$

$$= (e\,(g_1, g_2)^c M) \frac{e(g, (u' \prod_{i=v} v_i)^{rc})}{e(g_1, g_2)^c e((u' \prod_{i=v} v_i)^{rc}, g)}, \tag{12}$$

$$= M \tag{13}$$

Next the generated key is optimized using hybrid fragment horde bland lobo optimization algorithm.

3.3. Hybrid Fragment Horde Bland lobo Optimization

3.3.1. Traditional PSO Algorithm

Three vectors are used in the computations of the PSO algorithm. They are x-vectors, p-vectors, and v-vectors, accordingly. The p-vector (pbest) designates the place where the particle has found the object, and the x-vector tracks the particle's current location inside the search zone. best response to yet. Particle velocity, which forecasts where each succeeding particle will move throughout the course of that iteration, is likewise included in the v-vector. Initial random displacement of the particles occurs in preset directions. Gradually altering the particle's orientation allowed it to start moving on its own in the direction of the prior best position. Then it searches the area for the optimal spots to perform some fitness-related tasks, using the formula $fit = S^m - S$. Here, the particle's position is specified as $\vec{M} \in s^m$, Although its speed is given as $\vec{w}$. These two variables are initially chosen at random, and they are subsequently updated repeatedly using the two formulas in Equation (14)

$$\vec{w} = \omega\vec{w} + c_1 r_1\left(\vec{q} - \vec{M}\right) + c_2 r_2\left(\vec{f} - \vec{M}\right) \tag{14}$$

The inertia weight, or w in this instance, is a user-defined behavioural parameter that controls the degree of particle velocity recurrence. The particles implicitly interact with one another since the even before position of the particle (pbest position) isq, and the prior-best location of the particle inside the swarm (gbest position) is $\vec{f}$. The acceleration constants are c_1, c_2 and Utilizing the stochastic variables, this is graded. $r_1, r_2 \sim U(0,1)$. Equation illustrates that regardless of fitness gains, the particle is propelled to the next location in the search region by adding the velocity to its current position (15) as portrayed in Figure 6.

$$\vec{M} \leftarrow \vec{M} + \vec{w} \tag{15}$$

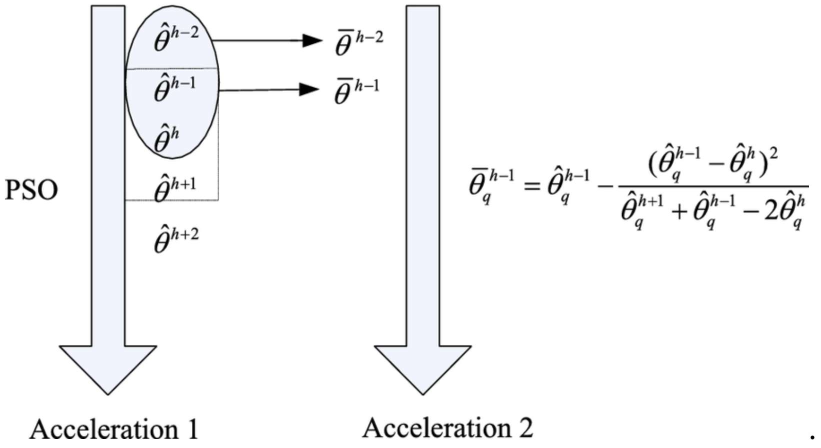


Figure 6. Architecture of Traditional PSO Algorithm.

3.3.2. Traditional GWO Algorithm

Hierarchical search agents exist in the GWO algorithm. Equations (16) and (17) use mathematics to illustrate how encircling occurs as grey wolves hunt their prey (17).

$$\vec{B} = \left|\vec{E}.\vec{M}_q(u) - \vec{M}(u)\right| \tag{16}$$

$$\vec{M}(u+1) = \vec{M}_q(u) - \vec{H}.\vec{B} \tag{17}$$

Where u the current iteration is handed to the term "coefficient vectors" is used to describe $\vec{H}$ and $\vec{E}$. Grey wolves have a special ability to locate their prey and surround it. Using the heightened awareness of potential prey sites of alpha, beta, and delta wolves, these grey wolf hunting behaviours are statistically replicated. Regardless of whether the further solutions are necessary, the top three are taken into account. Below are the mathematical equations (18) through (24):

$$\vec{B}_\alpha = \left|\vec{E}_1.\vec{M}_\alpha - \vec{M}(u)\right| \tag{18}$$

$$\vec{B}_\beta = \left|\vec{E}_2.\vec{M}_\beta - \vec{M}(u)\right| \tag{19}$$

$$\vec{B}_\delta = \left|\vec{E}_3.\vec{M}_\delta - \vec{M}(u)\right| \tag{20}$$

$$\vec{M}_1 = \vec{M}_\alpha - \vec{H}_1.\left(\vec{B}_\alpha\right) \tag{21}$$

$$\vec{M}_2 = \vec{M}_\beta - \vec{H}_2.\left(\vec{B}_\beta\right) \tag{22}$$

doi:10.20944/preprints202404.0386.v1

$$\vec{M}_3 = \vec{M}_\delta - \vec{H}_3.\left(\vec{B}_\delta\right) \tag{23}$$

$$\vec{M}(u+1) = \frac{\vec{M}_1 + \vec{M}_2 + \vec{M}_3}{3} \tag{24}$$

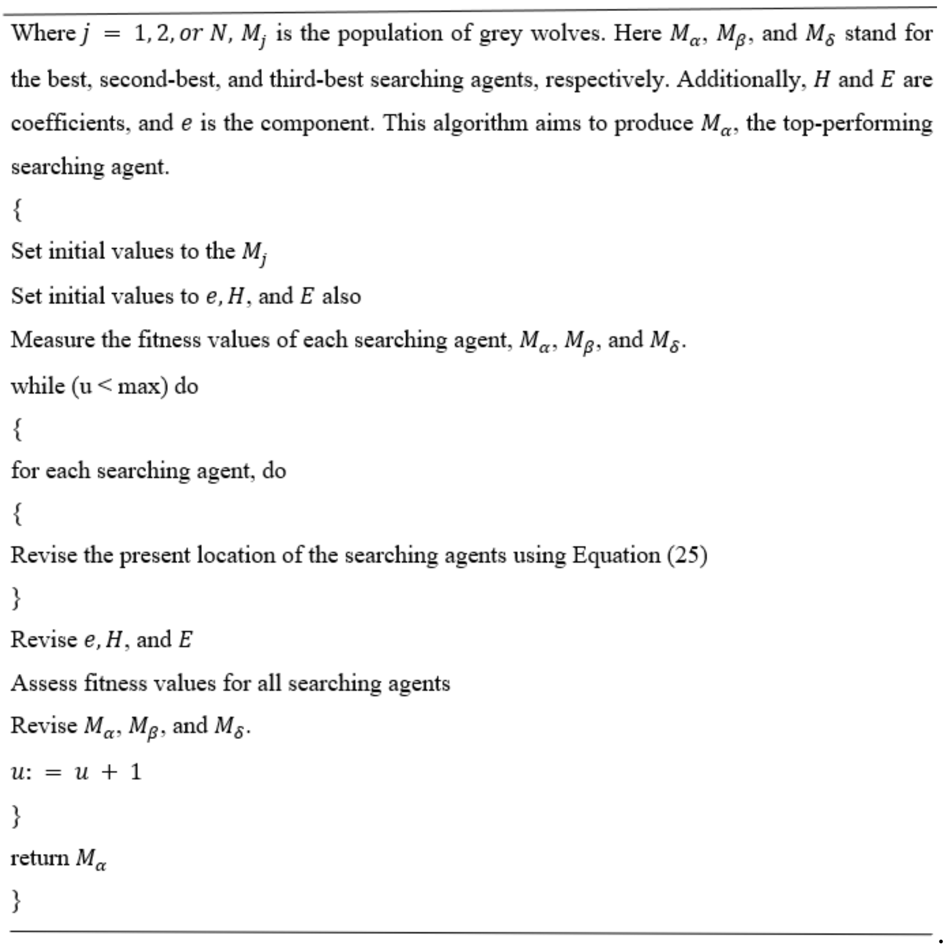
For despite their good performance, traditional algorithms can be improved upon to address their flaws and raise the bar.

3.3.2. Hybrid Fragment Horde Bland Lobo Optimization

Traditional algorithms can be improved upon despite having good performance to solve the shortcomings and raise the bar. There are some flaws in the conventional PSO algorithm, including inferior performance across a variety of fields. Along with these disadvantages, the GWO algorithm also has slower convergence, worse solution precision, and less effective local searching. Therefore, additional research is needed to enhance robustness and integration. These issues are dealt with in this work using a brand-new hybrid methodology. In the suggested Hybrid Fragment Horde Bland Lobo Optimization, the PSO method's criteria are blended with the GWO algorithm. Equations (15) and (16) illustrate the mathematical model of the prey enclosure in the proposed approach, whereas Equations (17) show the hunting strategy's mathematical model (18). The location has been updated, which is the main reformation in the recommended paradigm. Equation (19) illustrates the updating of the position in our Hybrid fragment horde bland lobo Optimization model, where $\vec{M}$ denotes the velocity for updating the location of PSO, as indicated in Equations (18) and (24).

$$M(u+1) = \frac{\vec{M}_1 + \vec{M}_2 + \vec{M}_3 + \vec{M}}{4} \tag{25}$$

Again, in the classic PSO method, c1 and c2 are regarded as acceleration constants, however in the recommended Hybrid fragment horde bland lobo Optimization model c_1 and c_2 fluctuate in accordance with the values 0.1, 0.3, 0.5, 0.7, and 1. The optimal key is generated using the Hybrid fragment horde bland lobo Optimization. Algorithm 1 presents the ideal key selection based on Hybrid fragment horde bland lobo Optimization.



Where $j = 1, 2, or\ N$, M_j is the population of grey wolves. Here M_α, M_β, and M_δ stand for the best, second-best, and third-best searching agents, respectively. Additionally, H and E are coefficients, and e is the component. This algorithm aims to produce M_α, the top-performing searching agent.

```
{
    Set initial values to the $M_j$
    Set initial values to $e, H$, and $E$ also
    Measure the fitness values of each searching agent, $M_\alpha$, $M_\beta$, and $M_\delta$.
    while (u < max) do
    {
        for each searching agent, do
        {
            Revise the present location of the searching agents using Equation (25)
        }
        Revise $e, H$, and $E$
        Assess fitness values for all searching agents
        Revise $M_\alpha$, $M_\beta$, and $M_\delta$.
        $u: = u + 1$
    }
    return $M_\alpha$
}
```

.

4. Results and Discussion

Java code for clustering and SecPri-BGMPOP technique was used in the trials, which were done under Windows utilizing HADOOP jars in cloudsim. The efficacy of the SecPri-BGMPOP approach is evaluated using information by using the link "https://www.hindawi.com/journals/bmri/2014/781670". From 1998 to 2008, medical management data were collected from 130 US facilities and coordinated delivery systems (10 years). Prior to the clustering operation, this dataset is further populated and prepared. Our research's experimental findings are contrasted with the current system. By tracking the amount of time and memory used during execution, the SecPri-BGMPOP technique approach is assessed.

4.1. Running Time

The whole SecPri-BGMPOP execution duration in Cloudsim is precisely calculated and measured in milliseconds (ms).

4.2. Memory Usage

The memory utilization was effectively organized using the Java-based CloudSim programme. Kilobytes are used to measure the size of the SecPri-memory BGMPOP's footprint and the stack memory needed for newly generated objects (kb).

4.3. Output measures

The Boost Graph convolutional network clustering approach's The SecPri-BGMPOP threshold is "1" for time and memory efficiency tests. Table 1 lists the results for various cluster sizes.

Table 1. Cluster Based Time, Memory Measures for SecPri-BGMPOP Threshold value "1".

Number of clusters	Time(sec)	Memory(KB)
1	31.569	1520
2	30.638	1613
4	30.456	1663

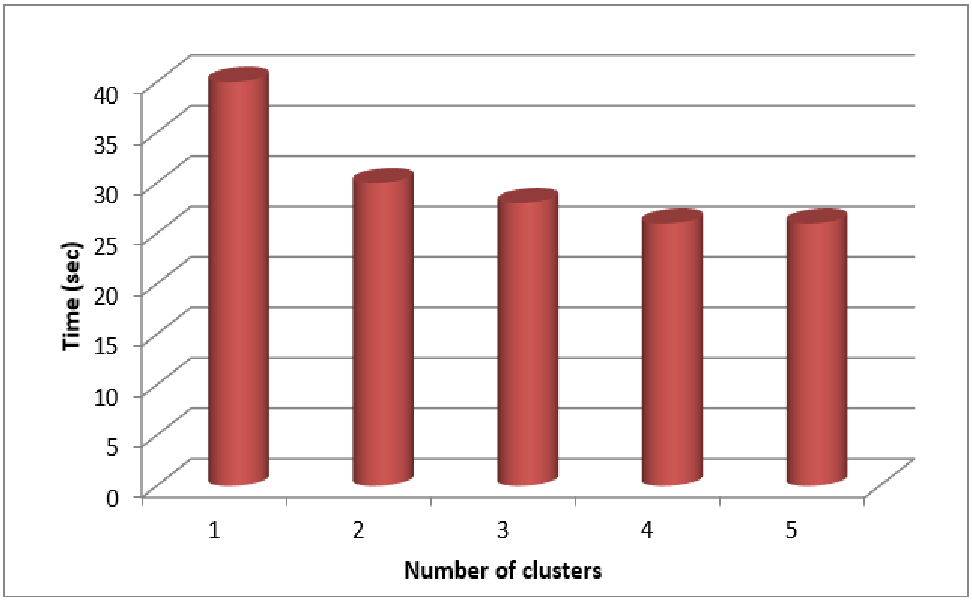


Figure 7. Clustering based Time Measure for SecPri-BGMPOP Threshold value "1".

According to a graph drawn for time against cluster count (Figure 7), the process takes less time to complete the more clusters there are. Because many mappers can run simultaneously, the procedure can be completed more quickly plotted as shown in Figure 4, and it is discovered that the least amount of memory is needed to complete a process when the number of clusters is low because higher cluster counts also result in higher memory allocation.

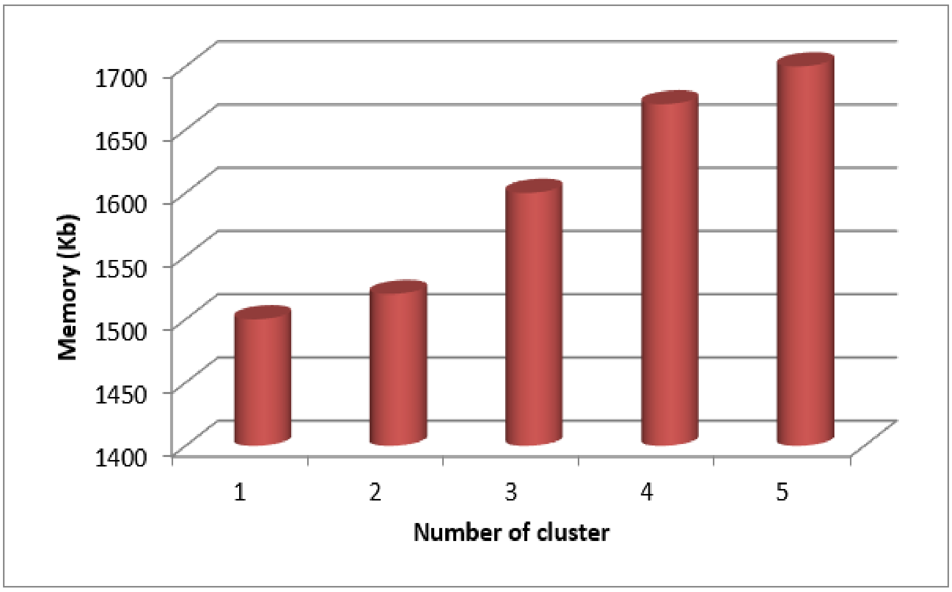
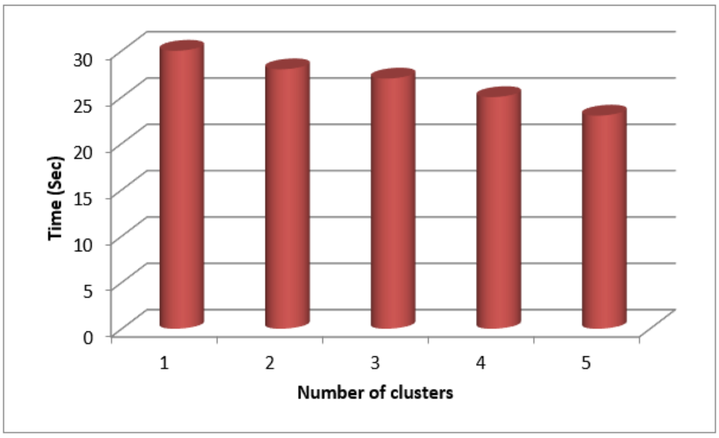


Figure 8. Cluster based Memory Measures for SecPri-BGMPOP Threshold value "1"

Table 2. Clustering Based Time and Memory Measures for SecPri-BGMPOP Threshold value "2".

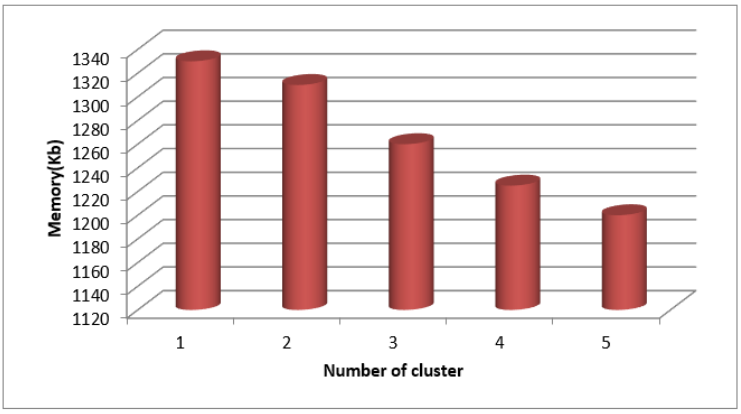
Number of clusters	Time(sec)	Memory(KB)
1	29.55	1325
2	27.55	1265
4	26.45	1232

Table-2 lists the clustering-based time and memory metrics for the "2" SecPri-BGMPOP threshold, and Figures 9 and 10 display graphs of these measures. The findings demonstrate that when the number of clusters rises, both time and memory are decreased since the threshold chosen is the best option.



.

Figure 9. Clustering based Time Measure for SecPri-BGMPOP Threshold "2".



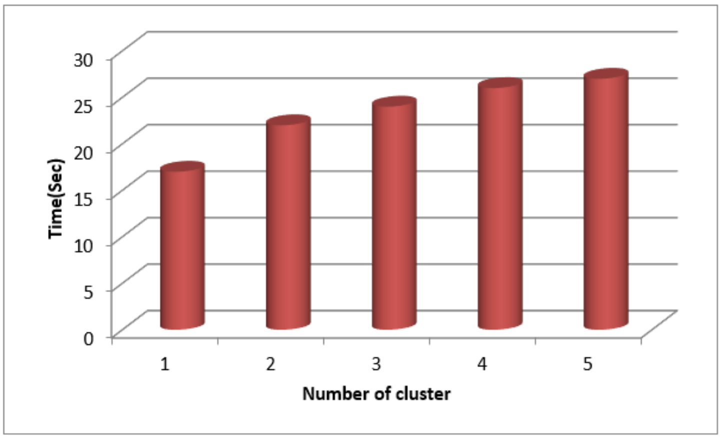
.

Figure 10. Clustering based Memory Measure for SecPri-BGMPOP Threshold value "2".

Table 3. Clustering Based Time and Memory Measures for SecPri-BGMPOP Threshold value "4".

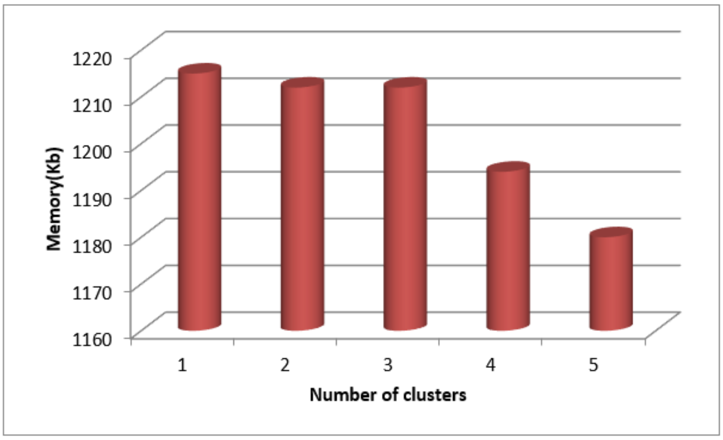
Number of clusters	Time(sec)	Memory(KB)
1	22.37	1216
2	24.45	1216
4	25.64	1199

Table 3 lists the findings of the clustering-based time measurements for SecPri-BGMPOP threshold value 4 for WFCM. The graphs in Figures 10 and 11 show that when the cluster size increases, the time to complete increases since a high threshold requires more processing power and requires less memory.



.

Figure 11. Clustering based Time Measure for SecPri-BGMPOP Threshold value "4".

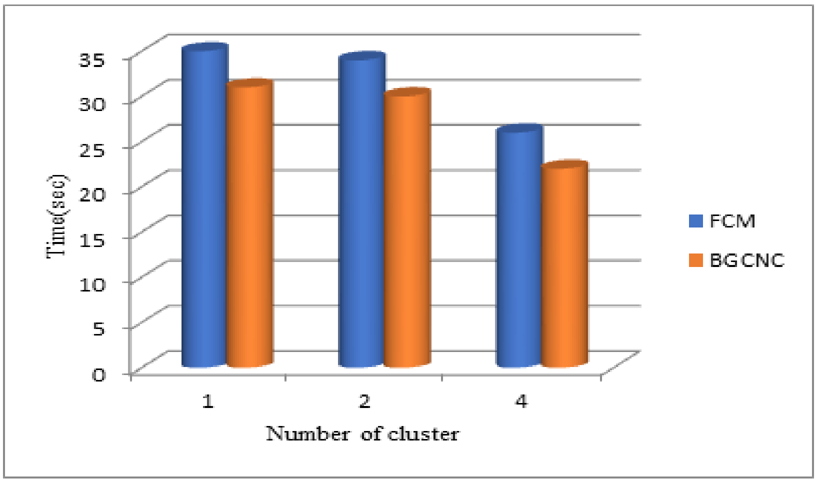


.

Figure 12. Clustering based Memory Measure for SecPri-BGMPOP Threshold value "4".

4.4. Performance Comparison

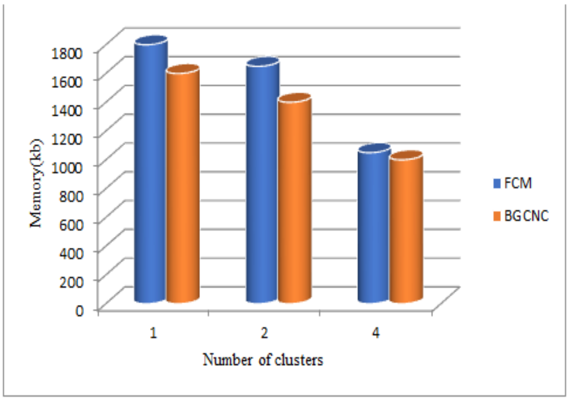
Based on clusters 2, 3, and 4, the Boost Graph convolutional network clustering method was evaluated in terms of time and memory for the SecPri-BGMPOP threshold values of 1, 2, and 4. The suggested system and the Boost Graph convolutional network clustering used in earlier work are then compared in terms of time and memory requirements. The Boost Graph convolutional network clustering recommended method produced improved outcomes, as illustrated in Figure 13.



.

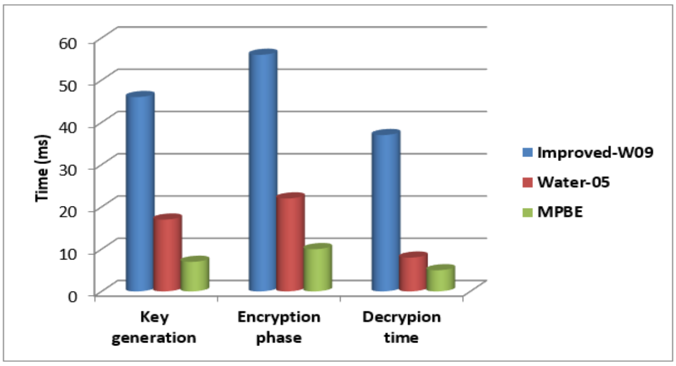
Figure 13. Comparison of Time Measures with BGCNC and FCM for SecPri-BGMPOP threshold value "1 – 2 -4".

When comparing the Boost Graph convolutional network clustering(BGCNC) -based SecPri-BGMPOP method to FCM clustering, it is discovered that Boost Graph convolutional network clustering performs better. Here, we've examined time and memory across various cluster sizes. When we examine the data, we discover that the proposed system uses less memory and requires less time than the current FCM-based technique (see Fig.10). As a result, our suggested Boost Graph convolutional network clustering approach performs better and produces better results.



.

Figure 14. Comparison of proposed Vs FCM Memory Measures for SecPri-BGMPOP threshold value "1, 2, 4".



.

Figure 15. Timing comparison graph for the proposed work in key generation, encryption phase and decryption phase.

doi:10.20944/preprints202404.0386.v1

The amount of time needed for computation in the key generation phase using the proposed MPBE, upgraded W09, Water 05, and MPBE curve. When compared to other methods, it can be seen that the Bibber approach requires less calculation time for the key creation, encryption, and decryption phases.

Table 4. Sanitization method and proposed SecPri-BGMPOP.

Performance metrics	Proposed versus previous approaches	
	Sanitization method	**SecPri-BGMPOP**
Information loss	0.07%	0.024%
Throughput	3.5Mbps 3.625Mbps	7Mbps 7.16Mbps
Encryption time	0.11s	0.0086s
Decryption time	0.054s	0.315s
Efficiency	46.87s	58.130s

Table 5. Performance enhanced Hybrid fragment horde bland lobo Optimization.

HFHBLO	PSO	GWO	Attack
Better than	**0.26%**	**0.23%**	**CCA**
Superior to	**0.40%**	**0.29%**	**CPA**

The simulation thus shows that our suggested information security technique outperformed the currently used traditional algorithms based on particular assaults. As a result, it is evident from the simulation results that our sanitising strategy outperforms other standard algorithms currently in use. Protecting this type of data is essential since it is necessary to use sensitive diagnostic information about autism to determine if a person is autistic or not. This type of information is more relevant in the healthcare industry. This study's findings demonstrated that our suggested cleaning method protects these data against some threats more effectively than existing techniques. Nonetheless, it is indicated that, in terms of data security and privacy, the healthcare industry can make extensive use of our suggested strategy.

5. Conclusions

SecPri-BGMPOP strategy aims, security and privacy protection have been the subjects of distinct research endeavours in the past. The purpose of this essay was to provide big data analytics with security while maintaining privacy. Initially Boost Graph Convolutional Network Clustering (BGCNC) algorithm, which reduces computational complexity in terms of time and memory measurements, is first applied to the input dataset to begin the clustering process. Second, enlarge by employing a piece of the magnifying bit string to generate a safe key, pinpointing based encryption avoids amplify leakage even if a rival or attacker decrypts the key or asymmetric encryption. Finally, to determine the accuracy of the method, an optimal key was created using a meta-heuristic algorithmic framework called Hybrid Particle Swarm and Grey Wolf Optimization (HPSGWO). Our proposed method currently kept in a cloud environment, allowing analytics users to utilize it without risking their privacy or security. We compared the technique based on Boost Graph convolutional network clustering to the current FCM approach and created Boost Graph convolutional network clustering to show that it has delivered superior results. Examining memory usage and execution time is a part of measuring performance. Confirming the experiment's results, it was found that the SecPri-BGMPOP strategy significantly cuts down on memory usage and computation time. We intend to present additional real-world uses for the SecPri-BGMPOP scheme in the future and contrast it with other encryption techniques. In order to accelerate the encryption and decryption processes even more, we also aim to improve the SecPri-BGMPOP scheme.

Author Contributions: MK. Conceptualization; JPA. initial draft of manuscript; GHS. Experimental design and Methodology; PKB: Validation and Analyses; JS. Review; WCL. Editing., Software. All authors have read and agreed to the published version of the manuscript.

Funding: Department of Electrical Engineering, Ming Chi University of Technology, Taiwan.

Institutional Review Board Statement: Not Applicable.

Informed Consent Statement: For this type of analysis formal consent is not needed.

Data Availability Statement: The authors do not provide supplementary data and material.

Acknowledgments: None.

Conflicts of Interest: The authors declare that they have no potential conflict of interest.

References

1. Lv, Z.; Qiao, L. Analysis of healthcare big data. *Future Gener. Comput. Syst.* **2020**, *109*, 103-110.
2. Yang, Yang, Xianghan Zheng, Wenzhong Guo, Ximeng Liu, and Victor Chang. "Privacy-preserving smart IoT-based healthcare big data storage and self-adaptive access control system." *Information Sciences* 479 (2019): 567-592
3. Hathaliya, J.J.; Tanwar, S. An exhaustive survey on security and privacy issues in Healthcare 4.0. *Comput. Commun.* **2020**, *153*, 311-335.
4. Chang, Victor, Yen-Hung Kuo, and Muthu Ramachandran. "Cloud computing adoption framework: A security framework for business clouds." *Future Generation Computer Systems* 57 (2016): 24-41.
5. Premkamal, P.K.; Pasupuleti, S.K.; Alphonse, P. J. A. A new verifiable outsourced ciphertext-policy attribute based encryption for big data privacy and access control in cloud. *J. Ambient Intell. Humaniz. Comput.* **2019**, *10*, 2693-2707.
6. CHELLADURAI, M.U.; Pandian, S.; Ramasamy, K. A blockchain based patient centric electronic health record storage and integrity management for e-Health systems. *Health Policy Technol.* **2021**, *10*, 100513.
7. Ogu, E.C. Cybersecurity for eHealth: A simplified guide to practical cybersecurity for non-technical healthcare stakeholders & practitioners. *Routledge*, **2022**.
8. Razzaq, A. Blockchain-based secure data transmission for internet of underwater things. *Clust. Comput.* **2022**, 1-20.
9. Liu, J.; Liu, Z.; Sun, C.; Zhuang, J. A data transmission approach based on ant colony optimization and threshold proxy re-encryption in wsns. *J. Artifi. Intell. Technol.* **2022**, *2*, 23-31.
10. Wang, T.; Ke, H.; Zheng, X.; Wang, K.; Sangaiah, A.K.; Liu, A. Big data cleaning based on mobile edge computing in industrial sensor-cloud. *IEEE Trans. Industr. Inform.* **2019**, *16*, 1321-1329.
11. Hosseinioun, P.; Kheirabadi, M.; Tabbakh, S.R.K.; Ghaemi. R. A new energy-aware tasks scheduling approach in fog computing using hybrid meta-heuristic algorithm. *J. Parallel Distrib. Comput.* **2020**, *143*, 88-96.
12. Shukla, D.K.; Ali, S.; Trivedi, M.C. Energy Aware Scheduling of Tasks in Cloud environment. *Turk. Online J. Qual. Inq.* **2021**, *12*.
13. Papageorgiou, P. A Novel Framework for Maritime Security Assessments and Its Applications on the Shipping Industry. Cyber Security Examples. *PhD diss., Liverpool John Moores University (United Kingdom),* **2022**.
14. Miyachi, K.; Mackey, T.K. hOCBS: A privacy-preserving blockchain framework for healthcare data leveraging an on-chain and off-chain system design. *Inf. Process. Manag.* **2021**, *58*, 102535.
15. Pramanik, P.K.D.; Pal, S.; Mukhopadhyay, M. Healthcare big data: A comprehensive overview. *Research anthology on big data analytics, architectures, and applications,* **2022,** 119-147.
16. Riaz, S.; Khan, A.H.; Muhammad Haroon, Sadia Latif, and Sana Bhatti. "Big data security and privacy: Current challenges and future research perspective in cloud environment. *In 2020 International Conference on Information Management and Technology (ICIMTech),* **2020**, pp. 977-982.
17. Sauber, A.M.; El-Kafrawy, P.M.; Shawish, A.F.; Amin, M.A.; Hagag, I.M. A New Secure Model for Data Protection over Cloud Computing. *Comput. Intell. Neurosci.* **2021,** 1-11.
18. Samaraweera, G.D.; Chang, J.M. Security and privacy implications on database systems in Big Data era: A survey. *IEEE Trans. Knowl. Data Eng.* **2019**, 33, 239-258.
19. Amanullah, M.A.; Habeeb, R.A.A..; Nasaruddin, F.H.; Gani, A.; Ahmed, E.; Nainar, A.S.M.; Akim, NM.; Imran, M. Deep learning and big data technologies for IoT security. *Comput. Commun.* **2020**, *151*, 495-517.
20. Rai, M.; Mandoria, H. A study on cyber crimes cyber criminals and major security breaches. *Int. Res. J. Eng. Technol.* **2019**, *6*, 1-8.
21. Lv, Z., Wang, L., Guan, Z., Wu, J., Du, X., Zhao, H., Guizani, M. An optimizing and differentially private clustering algorithm for mixed data in SDN-based smart grid. *IEEE access*, **2019**, *7*, 45773-45782.
22. Shanmugapriya, E., Kavitha, R. Medical big data analysis: preserving security and privacy with hybrid cloud technology. *Soft Comput.* **2019**, *23*, 2585-2596.

doi:10.20944/preprints202404.0386.v1

23. Tian, Y., Wang, Z., Xiong, J., Ma, J. A blockchain-based secure key management scheme with trustworthiness in DWSNs. *IEEE Transactions Indus. Inform.* **2020**, *16*, 6193-6202.
24. Chen, J., Ramanathan, L., Alazab, M. Holistic big data integrated artificial intelligent modeling to improve privacy and security in data management of smart cities. *Microprocess. Microsyst.* **2021**, *81*, 103722.
25. Iqbal, S.; Kiah, M.L.M.; Zaidan, A.A.; Zaidan, B.B.; Albahri, O.S.; Albahri, A.S.; Alsalem, M. A. Real-time-based E-health systems: Design and implementation of a lightweight key management protocol for securing sensitive information of patients. *Health Technol.* **2019**, *9*, 93-111.
26. Elhoseny, M.; Haseeb, K.; Shah, A.A.; Ahmad, I.; Jan, Z.; Alghamdi, M.I. IoT solution for AI-enabled PRIVACY-PREServing with big data transferring: an application for healthcare using blockchain. *Energies* **2021**. *14*, 5364.
27. Ramachandra, M.N.; Rao, M.S.; Lai, W.C.; Parameshachari, B.D.; Babu, J.A.; Hemalatha, K.L. An Efficient and Secure Big Data Storage in Cloud Environment by Using Triple Data Encryption Standard. *Big Data Cogn. Comput.* **2022**, *6*, 101.
28. Gehlot, A.; Misra, N. Privacy and Security Enabling for Healthcare Data using Lightweight Deep learning with Cryptography. *In 2022 IEEE 2nd Mysore Sub Section International Conference (MysuruCon*), **2022**, pp. 1-6.
29. Mehbodniya, A.; Neware, R.; Vyas, S.; Kumar, M.R.; Ngulube, P.; Ray, S. Blockchain and IPFS integrated framework in bilevel fog-cloud network for security and privacy of IoMT devices. *Comput. Math. Methods Med.* **2021**.
30. Refaee, E.; Parveen, S.; Begum, K.M.J.; Parveen, F.; Raja, M.C.; Gupta, S.K.; Krishnan, S. Secure and scalable healthcare data transmission in IoT based on optimized routing protocols for mobile computing applications. *Wirel. Commun. Mob. Comput.* **2022**.
31. Senel, F.A.; Gökçe, F.; Yüksel, A.S.; Yi ˘ git, T. A novel hybrid PSO–GWO algorithm for optimization problems. Eng. Comput. **2018**, 1359–1373.
32. Zolghadr-Asli, B.; Bozorg-Haddad, O.; Chu, X. Crow Search Algorithm (CSA). In Advanced Optimization by Nature-Inspired Algorithms. Studies in Computational Intelligence; Bozorg-Haddad, O., Ed.; Springer: Singapore, 2018; Volume 720.
33. Alphonsa, M.M.A.; Amudhavalli, P. Genetically modified glowworm swarm optimization based privacy preservation in cloud computing for healthcare sector. Evol. Intell. **2018**, 11, 101–116.
34. Kavitha C; Anita X. Task failure resilience technique for improving the performance of MapReduce in Hadoop. Etri Journal 2020, 42, 748 -760.
35. C. Kavitha; S. R. Srividhya; Wen-Cheng Lai; Vinodhini Mani. IMapC: Inner MAPping Combiner to Enhance the Performance of MapReduce in Hadoop. Electronics 2022, 11, 1599 .

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