MPDP k-medoids: Multiple partition differential privacy preserving k-medoids clustering for data publishing in the Internet of Medical Things

Zekun Zhang¹, Tongtong Wu¹, Xiaoting Sun¹ and Jiguo Yu²,³

Abstract
The tremendous growth of Internet of Medical Things has led to a surge in medical user data, and medical data publishing can provide users with numerous services. However, neglectfully publishing the data may lead to severe leakage of user’s privacy. In this article, we investigate the problem of data publishing in Internet of Medical Things with privacy preservation. We present a novel system model for Internet of Medical Things user data publishing which adopts the proposed multiple partition differential privacy k-medoids clustering algorithm for data clustering analysis to ensure the security of user data. Particularly, we propose a multiple partition differential privacy k-medoids clustering algorithm based on differential privacy in data publishing. Based on the traditional k-medoids clustering, multiple partition differential privacy k-medoids clustering algorithm optimizes the randomness of selecting initial center points and adds Laplace noise to the clustering process to improve data availability while protecting user’s privacy information. Comprehensive analysis and simulations demonstrate that our method can not only meet the requirements of differential privacy but also retain the better availability of data clustering.

Keywords
Data publishing, differential privacy, k-medoids clustering, privacy preservation, Internet of Medical Things

Introduction
With the blossom of Internet of Things (IoT)¹–³ and 5G,⁴ intelligent mobile devices and wearable devices are rapidly spreading, and various applications are becoming more and more prevalent, such as social networks, e-commerce, location-based services (LBSs),⁵–⁷ and Internet of Medical Things (IoMT) services.⁸ IoMT is an emerging technology which connects both the reliability and safety of medical devices and dynamicity and scalability of IoT to improve the patient’s quality of life by enabling personalized e-health services without limitations on time and location.⁹

IoMT interconnects between not only numerous personal medical devices (i.e. medical wearable and

¹College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao, P.R. China
²School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan, P.R. China
³Shandong Laboratory of Computer Networks, Jinan, P.R. China

Corresponding author:
Jiguo Yu, School of Computer Science and Technology, Qilu University of Technology (Shandong Academy of Sciences), Jinan 250353, Shandong, P.R. China.
Email: jiguoyu@sina.com
implanted sensors) but also between devices and healthcare service providers, such as hospitals, medical researchers, or private companies. IoMT can provide a multitude of user data, which can be collected, mined, and published for various purposes. For example, they can predict and analyze the user’s habits, providing users with more accurate services.10

However, since these data contain private information that cannot be disclosed, once these data are maliciously used, it will cause unpredictable loss to the individual, and even cause extremely bad social influences.11 For example, in order to analyze the patients’ condition or the need for disease research work, the hospital collects the patients’ case information, and thus, the case information may be leaked during data mining. Most of the information is the patients’ privacy, neither hospitals nor patients want sensitive data leaked.

From the perspective of information disclosure, data publishing is the data holder’s disclosure of data information to other people or organizations.12 However, in the actual publishing process, once the publisher’s protection measures are not perfectly safe, the security of sensitive data cannot be guaranteed, and it may bring unpredictable security risks and privacy concerns to the individuals related to these data. Therefore, protecting the privacy of IoMT user’s data in data publishing is a great challenging issue.13

In recent years, more and more researchers have devoted themselves to how to improve the data privacy protection of data publishing applications.14,15 From the specific details, privacy protection has two important perspectives: one is to explore the privacy protection methods in data applications,16 and the other is how to handle data applications better.17 The identifier of the data set is the key information for determining the specific metadata. It can be used to accurately lock a specific piece of data. In this case, the data information can be protected by deleting the identifiers attribute so that attackers cannot match the data with specific individuals. Today’s data analysis is based on a large number of data sets, mining and accessing massive amounts of data. When a large amount of data is accessed at the same time, authentication and admission control cannot work out. Such methods can only prevent sensitive information from being direct interception, but when querying multiple data sets, there is no guarantee that the data is safe.

Existing privacy protection methods mainly involves three areas: data encryption, restricted publishing, and data distortion. Data encryption is a method of hiding information by means of encryption. It is used more in distributed systems, such as secure multi-party computing,18,19 which requires a large running memory. Restricted publishing is based on a specific environment to carefully select the scope and method of information publishing, such as data generalization and data anonymization, but the attacker can still reason jointly through the published data and data sets obtained from other places to get detailed information about a specific individual.20 Data distortion is a method that is specific to sensitive data. It employs some specific processing to make the original value of sensitive data no longer recognizable, but keeps other required attribute values unchanged so that it can be analyzed. For example, techniques such as adding noise, randomization, switching, and blocking add interference information to sensitive information, so that even if the leakage does not distinguish the specific value of the sensitive information, the disturbed data must still have statistical characteristics.

All these techniques have their specific advantages, but there are other drawbacks that cannot be avoided. It does not modify any data based on the encryption technique, so the validity and security of the data are perfectly guaranteed. However, it has a lot of computational overhead. The data distortion technique is more efficient, but there are still a lot of data losses. The limited publishing technique has its own advantage to ensure the authenticity of the published data, but at the same time, the published data will inevitably have some data loss.

Combining data clustering with privacy protection techniques in the data publishing process is an effective method to solve the problem of user privacy leakage. Wong et al.21 proposed the (α, k)-anonymity algorithm, which uses clustering methods to protect the relationship between nodes and their attributes, and pays more attention to personalized privacy protection claims. k-means is simple to implement and has a fast clustering speed. Blum et al.22 applied differential privacy to the k-means algorithm, but their algorithms are not robust enough for noise processing. Li et al.23 proposed another differential privacy improved differential privacy (IDP) k-means algorithm based on the initial center selection method. However, the clustering efficiency and accuracy of their algorithms are not high enough.

Dwork24 first proposed differential privacy (hereinafter referred to as differential privacy protection). Differential privacy has a specific analysis of the attack model with a strict prevention mechanism and has the advantages of mathematical calculations, which can quantitatively prove the specific privacy protection ability. Differentially privacy data publishing is a prevalent model that can release aggregate information to the public without disclosing any individual’s data.25

An important issue in privacy-protected data publishing in IMoT is how to make the right trade-off between user’s privacy and data availability. Most of the existing methods do not pay much attention to the availability of data when considering the privacy preservation. However, this approach may cause some
problems in practical applications, because in some practical scenarios, there is usually no clear privacy requirement, and people may be willing to accept more privacy loss to obtain greater data utility. In short, existing data publishing methods lack a comprehensive measurement of the privacy and availability of user data.

This article focuses on data publishing privacy protection in IMoT. To achieve both the privacy protection and the availability of user data in data publishing, this article proposes a novel system model for IoMT user data publishing and the multiple partition differential privacy \( k \)-medoids (MPDP \( k \)-medoids) clustering algorithm for data clustering analysis.

The main contributions of this article are as follows:

1. An IMoT data publishing model is proposed, which includes IMoT users, data centers, and data applications. Its main function is to realize the collection of user data, privacy protection, and data clustering and publishing.
2. A clustering privacy protection algorithm MPDP \( k \)-medoids based on differential privacy is proposed. First, to solve the problem of poor availability of data after clustering due to improper selection of the initial center point during the clustering process, the selection of the initial center point is optimized by multiple division of the data set, and a data center point selection algorithm is constructed. Then, it combines differential privacy technique in the clustering process, and applies the Laplacian method to add noise data to realize the protection of user privacy data.
3. We prove the correctness of our schema and perform extensive experiments to validate our algorithm. The results indicate that our algorithm is superior to other algorithms in terms of efficiency, accuracy, and privacy preservation effect.

The remainder of the article is organized as follows. Section “Related work” introduces the related work. Section “Preliminaries” describes the preliminaries. Section “System model” gives the system model. In section “Proposed algorithm,” the MPDP \( k \)-medoids privacy protection clustering algorithm in IMoT is presented. Section “Experimental results” elaborates the experimental results. Finally, section “Conclusion” gives the conclusions.

**Related work**

Privacy-protected data publishing based on clustering has attracted more attention. Langari et al.\(^ {26} \) proposed a combined anonymity algorithm based on \( k \)-means clustering to minimize information loss and protect user privacy in social networks. Dou et al.\(^ {27} \) proposed a privacy protection data aggregation algorithm based on primitive clustering, which can effectively reduce data traffic and improve data privacy. Song et al.\(^ {28} \) introduced a hierarchical clustering method based on a sensitivity level division algorithm, and proposed a sensitive \( k \)-anonymity model, which can effectively protect data privacy and improve data security and practicability. Yang et al.\(^ {29} \) proposed a differential privacy protection method based on the adaptive grid method by adding Laplacian noise in the clustering.

Guan et al.\(^ {30} \) combined the non-parametric Bayesian algorithm and differential privacy, and proposed a differential private clustering algorithm of an infinite Gaussian mixture model, which used the Laplace mechanism for data publishing. Rong et al.\(^ {31} \) proposed a differential private solution, which converts the clustering problem into a classification problem without clear guidance to cover up the data, and adds noise in the process for differential privacy protection. Zheng et al.\(^ {32} \) proposed a new privacy protection data sharing framework that allows data contributors to share content upon request, adopting differential privacy to ensure privacy preservation.

Although the practicability of the method of clustering the published heterogeneous data to generate anonymous data is obviously better than the method without considering the clustering, the security protocol is not considered in this process, and the high-dimensional data cannot be processed.

In recent years, the publishing of dynamic data has also received more and more attention. Yang et al.\(^ {33} \) proposed a differential privacy protection model that considers attribute correlation, which overcomes the high temporal and spatial complexity and low data practicability of traditional privacy protection methods. The same degree of differential privacy as in the past also improves the availability of data. Li et al.\(^ {34} \) proposed a method of principal component analysis to optimize and protect privacy data publish, which improves the privacy of published data and the accuracy and practicability of the data.

Wang et al.\(^ {35} \) proposed an improved Verifiable Dynamic Encryption with Ranked Search (VDERs) scheme to achieve the previous privacy, cutting off the link between the search token and the added document by adding a counter and updating the buffer. The experimental results show that the scheme supports more fine-grained verification and reduces the proof of generation. Wu et al.\(^ {36} \) proposed a multi-objective algorithm based on a grid method to find the best solution as a candidate for disinfection, which can significantly reduce the side effects of disinfection and speed up the calculation cost algorithm. In order to ensure privacy...
and security and protect patient health data, Wang and colleagues\textsuperscript{37} proposed a healthcare data system based on the IoT cloud, which combines forward privacy and verifiable searchable encryption methods, which can effectively protect patient privacy.

Cao et al.\textsuperscript{38} proposed an efficient algorithm for calculating privacy leakage, which converts the differential privacy mechanism into data publishing. To solve the privacy protection data publishing problem of linear dynamic network, Lu and Zhu\textsuperscript{39} designed a dynamic network system to protect the initial state of the target and the privacy of published data. Case studies show that the proposed technology is effective. To solve the problems of high computational complexity of dynamic data and publishing, poor data availability, and high noise, Kanga et al.\textsuperscript{40} proposed a dynamic publishing method and an optimization strategy based on dynamic similarity, which can protect the privacy of data with lower computational complexity and higher data accuracy.

Although these methods can solve the issue of dynamic data publishing problems and can reduce computational complexity and meanwhile improve accuracy, they may waste some privacy budget and the corresponding processing time cannot meet actual needs.

With the development of IoT, data publishing and privacy protection have gradually been applied to real application scenarios. Li et al.\textsuperscript{41} proposed a method of dynamically published vehicle trajectory data with privacy protection function under \((k, \Delta)\) security constraints, which uses generalization functions to cluster data under security constraints, effectively reducing computational costs. Ou et al.\textsuperscript{42} studied cross-IoT data publishing and proposed a correlation noise mechanism for cross-correlation data privacy, which satisfies Pufferfish’s privacy protection and achieves the best data practicability.

To protect the real-time data publishing of smart meters, Shateri et al.\textsuperscript{43} adopted a privacy measure based on information theory and designed a privatization mechanism by adding a minimum amount of distortion to the smart meter measurement to provide a target level of protection. Cai et al.\textsuperscript{44} proposed an intelligent cyber-physical system data publishing mechanism that can not only save energy but also protect data privacy with excellent performance. Ma et al.\textsuperscript{45} proposed a differential privacy protection mechanism to protect the real-time trajectory data publishing of vehicle, and adopted dynamic sampling method to process the trajectory data to guarantee the availability of the data. Considering the privacy, profit, and fairness of taxi company participants, Cai and Zheng\textsuperscript{46} proposed a privacy protection traffic sharing framework between taxi companies to protect sensitive information. Although these methods have achieved high prediction accuracy and low error in data publishing privacy protection, and improved data availability, the problem of excessive data calculation time has still not been solved, and further research is needed.

**Preliminaries**

**Basics of differential privacy**

Differential privacy has been widely used in IMoT data publishing in recent years because it can ensure that any published data between all the same input data sets has equal probability. And based on this feature, it is guaranteed that all output information is insensitive to individuals. In addition, even if the attacker has a certain background knowledge, the privacy of the record can be effectively protected.

Data publishing based on differential privacy is mainly a one-time complete publishing of all possible query results using differential privacy technology. The data set processed by differential privacy technology is not an accurate data set. Users can selectively perform data query according to their own needs, so that the published data can still be highly practical under the premise of achieving privacy protection. In this section, we will introduce some basic concepts of differential privacy used in this article.

**Definition 1 (adjacent data sets).** There are two data sets \(D\) and \(D'\).\textsuperscript{47} If their biggest difference is that except for one record, the other structural attributes are the same, namely, \(|D\Delta D'| \leq 1\), then \(D\) and \(D'\) are called the adjacent data sets.

**Definition 2 (differential privacy).** There are two adjacent data sets \(D_1\) and \(D_2\), and there is at most one record difference between them.\textsuperscript{47} Given a random function \(M\), if all \(O\) is within the range of the value of function \(M\), that is, \(O \subseteq \text{Range}(A)\), there is

\[
Pr[M(D_1) \in O] \leq exp(e) \times Pr[M(D_2) \in O] \tag{1}
\]

holds.

It can be said that the algorithm \(M\) provides \(e\)-differential privacy protection, where \(Pr[\cdot]\) represents the disclosure risk of the event, which is controlled by the value of the random function \(M\), \(e\) is called the privacy protection budget parameter, and the value of \(e\) directly affects the level of privacy protection.

**Definition 3 (sensitivity).** The sensitivity of a function refers to the maximum change in the output result of the function when a function adds or deletes a record to the data set.\textsuperscript{48} For any query function \(f : D \rightarrow \mathbb{R}^d\), the input is a data set, and the output is the \(d\)-
Definition 5 (Euclidean and Manhattan distance). Let $A$ and $B$ be the two points in $n$-dimensional space, respectively, denoted by $A(x_1, x_2, \ldots, x_n)$ and $B(y_1, y_2, \ldots, y_n)$. The Euclidean distance $L_1$ and the Manhattan distance $L_2$ between $A$ and $B$ are defined as

\[
L_1 = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + \cdots + (x_n - y_n)^2} = \sqrt{n \sum_{i=1}^{n} (x_i - y_i)^2}
\]

\[
L_2 = |x_1 - y_1| + |x_2 - y_2| + \cdots + |x_n - y_n| = \sum_{i=1}^{n} |x_i - y_i|
\]

The main disadvantages of the $k$-means algorithm are as follows: (1) Since the cluster center is the centroid (that means finding an average for all points to get the cluster center point), the center point is likely not a point in the data set. It is likely that the class center does not have a specific physical meaning. (2) The $k$-means algorithm uses the $L_1$ distance function and is easily affected by noise data. In the calculation method of $L_1$, there is a square term, which will amplify the influence of noise data on the category center point, and will bias the category center point. The $k$-medoids algorithm has two improvements: (1) restricted cluster centers must come from data points and (2) use the $L_2$ distance function.

**k-Medoids algorithm**

The $k$-medoids algorithm is a variant of the $k$-means algorithm, and its biggest difference from the $k$-means algorithm is the selection of the cluster center point. The principle of the center point selected by the $k$-medoids algorithm is that the sum of the distances from all other points in the current cluster to the center point is the smallest.

**k-Medoids algorithm process.**

1. Step 1: randomly select $k$ points in the data set as the category center points.
2. Step 2: assign each data point to the nearest center point ($L_2$ distance function).
3. Step 3: for each category, redefine the category center point:
   1. Step 3.1: calculate the $L_2$ distance of each point in the category to all other points separately, and then sum these distances.
   2. Step 3.2: use the point with the smallest sum of distances to all other points as the category center point.

**Differential privacy $k$-medoids clustering**

The existing differential privacy $k$-medoids algorithm (DP $k$-medoids) for user data clustering, the specific steps are as follows:

1. Step 1: randomly select $p$ points as initial cluster center points, and add Laplace noise to each center point.
2. Step 2: calculate the distance between each point in the data set and the center point, and assign this point to the cluster from its nearest center point to form $k$ clusters.
3. Step 3: in each cluster, points are successively selected to calculate the consumption incurred by replacing the original central point with this point, and a point that consumes less than the original center point is selected as a new center point, and adding noise to this point.
4. Step 4: repeat Step 2 and Step 3 until the number of iterations reaches the threshold.

**System model**

The system model we proposed is mainly divided into three core parts: IoMT users, service providers, and third-party service providers. The specific system model is shown in Figure 1.

1. IoMT users: IoMT users will generate a lot of user information while using mobile IoMT devices, including name, age, occupation, phone number, home address, and so on. (At present, most IMoT APPs require users to perform real-name authentication. To some extent, real-name authentication registration also increases the risk of user privacy disclosure.) These information is uploaded to cloud servers provided by
specific service providers through various terminal devices.

2. Service provider: the service provider will obtain the original data from the cloud server that stores the user data of the IMoT, and then mine a large amount of user data according to different business needs. After the cloud server obtains the raw data through the query function, the data are clustered by our MPDP k-medoids algorithm and a small amount of noise is added to the center point data. Then, the service provider analyzes whether the data after adding noise meets the privacy protection requirements. If not, it continues to add noise, cluster, and analyze data until it meets the differential privacy requirements. Finally, the noise-added data will be published. At the same time, since each query of data consumes a privacy budget, the number of queries is counted when the third party queries the data, and the data set is updated before the specified threshold is reached to ensure the durability of privacy protection.

3. Third-party service acquirers: the third-party service acquirers may be hospitals, medical researchers, or private companies who obtain the noise-added data. Although these data are different from the raw data, they do not affect the use of the data. Therefore, they can obtain the required results after analysis and processing.

**System privacy analysis**

In this section, we analyze the security of the system model for IMoT data publishing. We assume that the service providers and third-party service providers in the system are not trusted. The data published by the service provider should not only meet the needs of the service acquirer but also ensure that the user’s private information is not leaked. We next analyze the privacy of each part of the model:
1. Service provider: social network users located in different regions will generate a large amount of personal data and upload it to the cloud server, and the service provider will conduct behavior analysis and model construction of the data. Therefore, in order to prevent the service provider from identifying the user’s identity through certain background knowledge, we add a small amount of noise to the data before data processing. In this way, even if the service provider obtains a certain background knowledge, it is difficult to obtain the user’s private information from the data.

2. Third-party service acquirers: a smart attacker may pretend to be a third-party service acquirer and attempt to attack the acquired data with a certain background knowledge. In general, they are unlikely to contact users and service providers directly. After the personal data are processed by the MPDP k-medoids algorithm, the attacker can only obtain the data after adding noise. Even if they obtain some user information by attacking other data sources, they cannot identify individual differences through background knowledge, so they cannot infer the user’s real data information.

3. Differential privacy: in our model, all the data generated by clustering will add Laplace noise. No one can obtain the specific amount of noise added, so it is impossible to obtain user data by eliminating noise. Therefore, user data privacy is ensured under the premise of satisfying effective data query. In addition, the privacy budget parameters in differential privacy can balance the privacy and availability of user data during the query process. So, in the process of clustering, all data attributes are treated as quasi-identifiers. Then, we add Laplace noise to the data in cluster and disturb the data to record the true value, so as to meet the privacy requirements of the differential privacy protection model. Compared with the conventional Laplace mechanism, since the data are also anonymized after clustering, the sensitivity of the query function can be reduced in the clustering operation. The reduced sensitivity leads to a reduction in the amount of added noise, which greatly enhances data availability.

Proposed algorithm
This section first introduces the motivation of MPDP k-medoids privacy protection clustering algorithm in the process of publishing IMoT data, then gives the algorithm, and finally elaborates time complexity analysis.

Algorithm motivation
Aiming at the problem that the clustering results of the traditional k-medoids algorithm are susceptible to the initial center point and the privacy leakage during the clustering process, we optimized the selection of the initial center point by dividing the data set multiple times. At the same time, we also better solve the problems of strong similarity of elements in the class and large differences between classes. At the same time, we also use differential privacy technology to add noise to the center point selected during each clustering process to ensure that privacy information does not leak during the clustering process.

MPDP k-medoids algorithm
For the data mining process of IMoT data publishing, we propose the MPDP k-medoids method based on differential privacy protection. The improvement of the method in selecting the center point is divided into two aspects: one is to select the initial center point and perform subset division, and the other is to use the optimized center point for clustering. The specific steps are as follows:

Step 1: select the center point to divide the subset.
1. Select a random initial center point, run the k-medoids algorithm on the original data set \( D \), and divide the original data set \( D \) into \( j \) subsets \( D = \{d_1, d_2, \ldots, d_j\} \). Then, continue to execute the k-medoids algorithm on each subset.
2. Divide \( f \) clusters into each subset, so that \( j \) subsets will have \( j'f \) clusters, and store them in \( J \). Calculate the distance from each point in each cluster in \( J \) to the center point, select the point with the smallest distance as the initial center point, and store it in \( S \).
3. Calculate the sum of squares of each center point to each other center point in \( S \), and select the point with the largest distance, which is the point farthest from the center point, and store it in \( R \).
4. Calculate the number of points in \( R \). If it is equal to \( j \), Step 1 is end. Otherwise, proceed to \( e \).

Step 2: use the center points generated in Step 1 for clustering.
1. Perform the second algorithm using the center point generated in Step 1, and calculate the distance from each point in \( D \) to the center point, and assign to the nearest center point to form \( j \) clusters.
2. Calculate the distance from each point to all other points in the cluster, select the distance
and the smallest point as the new center point, and add noise to the center point.

3. Repeat $e$ and $f$ until the point-to-set partitioning no longer changes or reaches the specified number of iterations.

The meaning of the parameters and symbols involved in the algorithm is shown in Table 1.

The specific algorithm is as follows.

By executing Algorithm 1, we get the optimized center point, and then use this center point to execute Algorithm 2.

Among them, the distance between two points is calculated using the Euclidean distance

$$
\text{dist}(x, y) = \sqrt{\sum_{i=1}^{dm}(x_i - y_i)^2}
$$

**Time complexity analysis**

**Theorem 1.** The time complexity of Algorithm 1 and the time complexity of Algorithm 2 are: $O(j * f)$ and $O(j * n * t)$.

**Proof.** For a given data set $D$, the total number of records contained in $D$ is $n$. Compared with the original $k$-medoids algorithm, this algorithm adds an optimization process of selecting the initial center point for the clustering process, and adds differential privacy technology to protect the privacy of the data set. The following will evaluate the time complexity of the algorithm, assuming the worst case.

The time consumed for Algorithm 1 is equal to the time complexity in dividing the elements into the set $S : O(j * f)$. For Algorithm 2, we suppose the number of iterations of the algorithm is $t$, and the time consumed by the algorithm during each clustering process is $O(n * j * t)$.

**Differential privacy proof**

In the following, we give the proof that Algorithm 2 meets differential privacy.

**Theorem 2.** According to the definition of differential privacy, MPDP $k$-medoids implements $\varepsilon$-differential privacy, which is

$$
Pr\{\text{MPDPK}(D1)\} \leq e^\varepsilon \cdot Pr\{\text{MPDPK}(D2)\}
$$

Among them, $D1$ and $D2$ are the two data sets that differ by only one record.

**Proof.** According to the definition of differential privacy, the random function $R$ provides $\varepsilon$-differential privacy.
Algorithm 2. Cluster and add noise to the data.

Input: D: initial data set, \( \{C_1, C_2, C_3, \ldots, C_j\} \): set of j center points, e: privacy budget parameter, p: number of clusters, and d: query dimension.
Output: \( \{C_1, C_2, \ldots, C_j\} \): Divided clusters.
1: \( A = \emptyset \), \( C = \emptyset (1 \leq i \leq j) \)
2: while Point-to-set division no longer changes do
3: for \( x_i \) in D do
4: for \( t = 1 \) to \( j \) do
5: \( A_t = \text{dist}(x_i, C_t) \)
6: \( A = A \cup \{A_t\} \)
7: end for
8: find the minimum value \( A_t(1 \leq t \leq j) \) in A
9: divide the data point into a cluster formed by the closest center point
10: \( C_i = C_i \cup \{x_i\} \)
11: end for
12: for \( i = 1 \) to \( j \) do
13: update the point with the smallest distance from other points in each cluster to the new center point \( C_t \)
14: add noise to each dimension of the center point \( C_t \)
15: \( C_t = C_t + (\text{Laplace}(b), \ldots, \text{Laplace}(b)) \)
16: \( \text{Laplace}(b) = \exp(-|C_t, i|/\Delta k)(1 \leq t \leq d)/\|C_t\| \) is the value of the \( t \)th dimension of point \( C_t \)
17: end for
18: end while
19: return clustering results

privacy protection for the cluster center points. Therefore, the disclosure risks of the noise-added center points \( d1 \) and \( d2 \) satisfy the following formula

\[
Pr[R(d1)] \leq e^e \cdot Pr[R(d2)] \tag{7}
\]

In MPDPK(\( d1 \)) and MPDPK(\( d2 \)), assuming that the attacker obtains all information except the target information in the cluster and the center point after adding noise, then the attacker can obtain the exact value of a sensitive attribute by calculating the distance between the data sample and the center point. In MPDPK(\( d1 \)) and MPDPK(\( d2 \)), assuming that the attacker obtains all information except the target information in the cluster and the center point after adding noise, then the attacker can obtain the exact value of a sensitive attribute by calculating the distance between the data sample and the center point. The only variable that the attacker uses to obtain the accurate value of the target information is the center point, and the accuracy of the center point directly affects the accuracy of the target information. Therefore, the probability of obtaining accurate information of the data samples in the cluster is equal to the disclosure risk of the exact value of the center point, so the following formula holds

\[
Pr\{\text{MPDPK}(D1)\} = Pr\{R(d1)\} \tag{8}
\]

\[
Pr\{\text{MPDPK}(D2)\} = Pr\{R(d2)\} \tag{9}
\]

Therefore, from the formulas of (7)–(9), it can be concluded that the clustering result disclosure risk with \( d1, d2 \) as the center point satisfies the following formula

\[
Pr[\text{MPDPK}(d1)] \leq e^e \cdot Pr[\text{MPDPK}(d2)] \tag{10}
\]

This formula meets the definition of differential privacy. Therefore, MPDPK k-medoids can provide \( e \)-differential privacy protection.

Experimental results

In this section, the MPDPK k-medoids algorithm proposed in this article is experimentally verified, and the feasibility of the algorithm is evaluated by comparing with the existing methods. Experimental results show that our proposed privacy protection method is more effective in the process of data publishing.

Experimental environment and data sets

We analyze and illustrate the availability of the MPDPK k-medoids algorithm through specific data experiments. The experimental environment is Inter(R) Core(TM) i3-2328M CPU @2.20 GHz, 8G memory, Windows7 64-bit operating system, experimental implementation using Java language, and IDE development tool is Eclipse, version: Mars.2 Release (4.5.2). The data set used in the experiment is all from the UCI Knowledge Discovery Archive database. The specific information is shown in Table 2. \( D1 \) is the Conventional and Social Media Movies (CSM) data set containing conventional and social media features, which includes 217 pieces of tuple data and 12 attributes. \( D2 \) is the Activities of Daily Living (ADL) data set containing the ADLs performed by two users on a daily basis in their own homes. It includes 543 pieces of tuple data and 3 attributes. \( D3 \) is the News data set, which includes 6000 pieces of tuple data and 61 attributes. \( D4 \) is the Adult data set containing US Census data, which is widely used to protect the privacy of data anonymity. It includes 48,842 pieces of tuple data and 20 attributes such as name, gender, occupation, address, educational background, and disease in the data set, among which disease is a sensitive attribute. We conduct experiments on these data sets and verify the effectiveness of our proposed algorithm by comparing with some other anonymous model methods.

Evaluation index

The effectiveness of clustering was evaluated by CH. Assuming \( k \) is the number of subsets that the clustering algorithm divides the data set, then \( D = \{x_1, x_2, \ldots, x_n\} = \{C_1, C_2, \ldots, C_j\} \). Among them, \( 1 \leq i \leq k \), subset \( C_i \) is a subcluster of data set \( D \), \( N_i \) is the number of records of subcluster \( C_i \), \( c \) is the center point of data set \( D \), \( c_i \) is the center point of
subcluster \( C_i \), and \( \text{dist}(x,y) \) represents the distance between \( x \) and \( y \).

The evaluation index of CH clustering effectiveness is as follows

\[
\text{CH} = \frac{1}{n-k} \sum_{i=1}^{k} \sum_{j=1}^{n} \text{dist}^2(x_j, x_i)
\]

\[
\text{CH} = \frac{1}{n-k} \sum_{i=1}^{k} \sum_{j=1}^{n} \text{dist}^2(x_j, x_i)
\]

As can be seen from the above formula, the CH index measures the tightness within the cluster by calculating the squared sum of the distances between the points in the class and the cluster center, and the squared sum of the distances between the center points of the clusters and the data set to measure the separation of the data set. CH is the ratio of the two. Therefore, a larger CH value means that the clusters are more compact, the clusters are more discrete, and the clustering effect is better.

\( F \)-measure is a commonly used index for evaluating the availability of clustering. The two clustering results are calculated using \( F \)-measure. The calculation result is directly proportional to the effect of clustering. \( F \)-measure mainly calculates two parameters: accuracy and recall. This article uses calculation accuracy to measure mainly calculates two parameters: accuracy is directly proportional to the effect of clustering. The two clustering results are more compact, the clusters are more discrete, and the clustering effect is better.

\( F \)-measure is a commonly used index for evaluating the availability of clustering. The two clustering results are calculated using \( F \)-measure. The calculation result is directly proportional to the effect of clustering. \( F \)-measure mainly calculates two parameters: accuracy and recall. This article uses calculation accuracy to measure the impact of changes in different \( k \) values on the clustering results. Assuming that the data sets \( D_1 \) and \( D_2 \) are clustered, the clustering results are \( D_{11}, D_{12}, \ldots, D_{1k}, D_{21}, D_{22}, \ldots, D_{2k} \), where \( 1 \leq i \leq k, 1 \leq j \leq k \), and the clustering results of the \( D_2 \) data set are used as comparison results, then \( D_{1i} \) and \( D_{2j} \) are accurate and the rate \( P \) is defined as

\[
P = \frac{\text{Precision}(D_{1i}, D_{2j})}{\left| D_2 \right|} = \frac{N_{ij}}{\left| D_2 \right|}
\]

where \( D_{1i} \) and \( D_{2j} \) are the clustering results of the two data sets, \( \left| D \right| \) represents the number of all elements in the data set \( D \), and \( N_{ij} \) represents the number of \( D_{1i} \) elements in \( D_{2j} \).

Furthermore, the size of the privacy budget directly affects the level of privacy protection. Currently, data publishing based on differential privacy mainly includes interactive frameworks and non-interactive publishing frameworks. In the interactive framework, when a user makes a query, the data publisher adds noise to the real result according to the designed privacy protection algorithm and returns the query result to the user. However, each query needs to consume a privacy budget, and when the privacy budget is exhausted, the differential privacy protection requirements will not be met. One way is to allocate privacy budgets. Another method is to fix the privacy budget, which is to set a threshold and update the data set before the threshold is reached. In the non-interactive framework, the data publisher adds noise to the raw data according to the set privacy protection algorithm to form synthetic data for users to query after publication.

Since medical data contain a lot of sensitive information, such as identity information, disease, income, and family medical history, privacy budgets are often set relatively small in our MPDP \( k \)-medoids medical data publishing. However, the smaller the privacy budget, the worse the availability of data, which can prevent attacks well for attackers, but it is disadvantageous for legitimate third parties, such as researchers of disease prevention and treatment programs. Therefore, in this article, we adopt a fixed privacy budget method and set a threshold for the number of queries at the same time, which is convenient for updating the data set in time to achieve the purpose of privacy protection.

### Analysis of experimental results

In order to analyze the effect of different \( j \) values on the experimental results, we observe the changes of the clustering results under the conditions of \( j = 3, j = 5 \), and \( j = 12 \). We adopt the \( k \)-medoids algorithm without any privacy protection to calculate the clustering results with accuracy and compare with other privacy protection algorithms.

Table 3 shows the results of executing each clustering algorithm under different \( j \) values for the \( D_2 \) and \( D_3 \) data sets, and calculating the accuracy with the clustering results of the traditional \( k \)-medoids algorithm.

From the above results, it can be seen that when the privacy budget is determined, the number of clusters forming different clusters has little effect on the results, but the accuracy rate is slightly reduced, mainly because as the number of clusters increases, the distance between the elements in the same cluster becomes smaller, and the boundary between the clusters becomes smaller. Since the privacy protection method used is performed by adding noise, it will affect the center point of the cluster, which will cause the distance between clusters. The larger the error, the lower the accuracy.

We use MPDP \( k \)-medoids algorithm and traditional privacy protection algorithm to cluster data in \( D_2 \) data set. The clustering results are shown in Figures 2–5.
The results of clustering by different algorithms are shown in Figures 2–5. Among them, different colors represent the different clusters formed by the cluster. From the figures, we can see that the effect of clustering of DP \(k\)-medoids and MPDP \(k\)-medoids compared with MPDP \(k\)-medoids is affected by the noise interference of random addition, while MPDP \(k\)-medoids algorithm optimizes the selection of the initial cluster center points and performs certain iteration optimization when adding random noise, so the final clustering effect is not disturbed by the added noise too much. Similarly, since IDP \(k\)-means also optimized the initial center point selection, the clustering effect of IDP \(k\)-means is also better than DP \(k\)-means, but the results of clustering compared to MPDP \(k\)-medoids are also partly affected by random noise.

In addition, we use MPDP \(k\)-medoids and traditional privacy protection algorithms to perform cluster analysis on each data set. In order to avoid the

| Data set | Privacy | \( j \) value | Accuracy IDP \( k \)-means | Accuracy MPDP \( k \)-medoids |
|----------|---------|---------------|---------------------------|-----------------------------|
| \( D_2 \) | 3       | 3             | 98.6%                     | 98.9%                       |
|          | 5       | 97.3%         | 97.5%                     | 98.1%                       |
|          | 12      | 95.5%         | 95.3%                     | 96.1%                       |
| \( D_3 \) | 2       | 3             | 99.1%                     | 99.0%                       |
|          | 5       | 97.2%         | 97.0%                     | 97.5%                       |
|          | 12      | 97.5%         | 97.3%                     | 97.7%                       |

DP: differential privacy; MPDP: multiple partition differential privacy.
influence of the generated Laplace random noise on the experimental results, we take the average value of CH for 30 times in different data sets for comparison.

The experimental results are shown in Figures 6–9. It can be seen that as the value of $\varepsilon$ increases, the value of CH also increases as a whole. This shows that we make the result of clustering greatly improved by optimizing the selected initial center points. Moreover, when the $\varepsilon$ value is large and the added noise is small, the availability of clustering is basically the same as the original clustering algorithm. Moreover, it can be seen from the comparison of $D_4$ and $D_2$ and $D_1$ data sets that the experimental results are still stable when the data attribute quantity and data quantity are large, which fully indicates that the MPDP $k$-medoids algorithm can still be very good in the case of large data quantity to ensure the high availability and effectiveness of clustering.

It can be seen from the experimental results that MPDP $k$-medoids algorithm to a certain extent guarantees that personal privacy will not be leaked, and also ensures the effectiveness of clustering, and it will not be greatly affected by adding Laplace noise. In Figure 6, when $\varepsilon$ is about 0.8, the effectiveness of the noise-added clustering starts to stabilize, and from the meaning of the ordinate, our algorithm is better than the other three clustering effectiveness after differential privacy protection. The algorithm can be seen from Figures 6–9 that the larger the size of the data set, the smaller the $\varepsilon$ is needed, indicating that the large data set needs to achieve better utility at a higher level of privacy protection. Because when adding the Laplace noise with sensitivity $d$, the parameter that determines the noise is the dimension of the data set, so the amount of noise added has nothing to do with the size of the data set.
Comparing Figures 6 and 9, we can see that the clustering availability of small data sets is lower than that of large data sets. According to Figures 6–9, we can also see that DP $k$-means and DP $k$-medoids clustering in small-scale data sets is effectively affected by noise points, but as the data scale increases, the effectiveness of clustering is not much different from the other two algorithms.

In addition, we know that the random noise added by differential privacy has no certain relationship with the size of the data set, so the MPDP $k$-medoids algorithm has certain advantages when processing a certain size data set. With the increase in the scale of the data set, the anti-noise ability of the algorithm is also stronger, and the performance will also be better in practical applications.

**Conclusion**

In recent years, the application of IoMT data mining has become more and more extensive, and the knowledge mined in a large amount of information has become more and more accurate. Attackers have been able to obtain personal privacy data from massive information through data mining. This article focuses on the privacy protection issues in clustering analysis in IoMT data publishing. We propose a novel system model for IMoT user data publishing with privacy preservation. Our model adopts MPDP $k$-medoids clustering algorithm for data clustering analysis to ensure the security of user data. Based on differential privacy and $k$-medoids clustering algorithm, we propose MPDP $k$-medoids clustering algorithm.

A further study of the problem is to pursue higher availability of results while ensuring privacy protection as much as possible. At the same time, because the amount of noise added by differential privacy protection is independent of the size of the data set, how to reduce the noise interference of the data set with less data volume, how to reduce the influence of input parameters on the clustering result, and how to balance the impact of adding noise and the accuracy of clustering are the main concerns in the future.

**Declaration of conflicting interests**

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

**Funding**

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Natural Science Foundation (NSF) of China under grant nos 61771289 and 61832012, the Humanity and Social Science Fund of the Ministry of Education under grant nos 20YJAZH078 and 20YJAZH127, the Open Project of Tongji University Embedded System and Service Computing of Ministry of Education of China under grant nos ESSCKF 2019-06 and ESSCKF 2019-08, and the National Innovation and Entrepreneurship Training Program for College Students under grant no. 201910424014.

**ORCID iDs**

Zekun Zhang https://orcid.org/0000-0002-6727-7691
Tongtong Wu https://orcid.org/0000-0003-0153-7523
Xiaoting Sun https://orcid.org/0000-0001-9507-3082
Jiguo Yu https://orcid.org/0000-0001-6451-1158

**References**

1. Li R, Sturtivant C, Yu J, et al. A novel secure and efficient data aggregation scheme for IoT. IEEE IoT J 2019; 6: 1551–1560.
2. Ni L, Zhang J, Jiang C, et al. Resource allocation strategy in fog computing based on priced timed Petri nets. IEEE IoT J 2017; 4: 1216–1228.
3. Cai Z and He Z. Trading private range counting over big IoT data. In: 2019 IEEE 39th international conference on distributed computing systems (ICDCS), Dallas, TX, 7–10 July 2019, pp.144–153. New York: IEEE.
4. Bai X, Wang Z, Lei Z, et al. Target tracking for wireless localization systems with degraded measurements and quantization effects. IEEE Trans Indus Electron 2018; 65: 9687–9697.
5. Zhang J, Yuan Y, Wang X, et al. RPAR: location privacy preserving via repartitioning anonymous region in mobile social network. Security Commun Netw 2012; 2012: 6829326.
6. Wang J, Cai Z and Yu J. Achieving personalized k-anonymity based content privacy for autonomous vehicles in CPS. IEEE Trans Indus Inform 2019; 16(6): 4242–4251.
7. Ni L, Tian F, Ni Q, et al. An anonymous entropy-based location privacy protection scheme in mobile social
networks. EURASIP J Wireless Commun Netw 2019; 2019: 93–111.
8. Zhang J, Liu H and Ni L. A secure energy-saving communication and encrypted storage model based on RC4 for EHRI. IEEE Access 2020; 8: 38995–39012.
9. Arthur G, Youakim B, Bertrand M, et al. Internet of Medical Things: a review of recent contributions dealing with cyber-physical systems in medicine. IEEE IoT J 2018; 5(5): 3810–3823.
10. Cai Z, He Z, Guan X, et al. Collective data-sanitization for preventing sensitive information inference attacks in social networks. IEEE Trans Depend Secure Comput 2018; 15(4): 577–590.
11. Xing K, Hu C, Yu J, et al. Mutual privacy preserving \( k \)-means clustering in social participatory sensing. IEEE Trans Indus Inform 2017; 13: 2066–2076.
12. Wang Z, Pang X, Chen Y, et al. Privacy preserving crowd-sourced statistical data publishing with an untrusted server. IEEE Trans Mobile Comput 2019; 18: 1356–1367.
13. He Z, Cai Z, Yu J, et al. Cost-efficient strategies for restraining rumor spreading in mobile social networks. IEEE Trans Vehi Technol 2017; 66(3): 2789–2800.
14. Xu L, Jiang C, Chen Y, et al. A framework for categorizing and applying privacy-preservation techniques in big data mining. IEEE Comput 2016; 49(2): 54–62.
15. Buczak AL and Guven E. A survey of data mining and machine learning methods for cyber security intrusion detection. IEEE Commun Survey Tutor 2016; 18(2): 1153–1176.
16. Zheng X and Cai Z. Follow but no track; privacy preserved profile publishing in cyber-physical social systems. IEEE IoT J 2017; 4(6): 1868–1878.
17. He Z, Cai Z and Yu J. Latent-data privacy preserving with customized data utility for social network data. IEEE Trans Vehi Technol 2017; 67(1): 665–673.
18. Furukawa J, Lindell Y, Nof A, et al. High-throughput secure three-party computation for malicious adversaries and an honest majority. In: Annual international conference on the theory and applications of cryptographic techniques, 2017, pp.225–255, https://eprint.iacr.org/2016/944.pdf
19. Wang X, Ranellucci S and Katz J. Authenticated garbling and efficient maliciously secure two-party computation. In: Proceedings of the 2017 ACM SIGSAC conference on computer and communications security, 2017, pp.21–37, https://acmccs.github.io/papers/p21-wangA.pdf
20. Kucisik A, Deisenroth MP, Peters J, et al. Model-based contextual policy search for data-efficient generalization of robot skills. Artif Intell 2017; 247: 415–439.
21. Wong RC-W, Li J, Fu AW-C, et al. (\( \alpha, k \))-anonymity: an enhanced \( k \)-anonymity model for privacy preserving data publishing. In: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining, 2006, pp.754–759, https://eprints.usq.edu.au/2094/1/Alpha-KDD06.pdf
22. Blum A, Dwork C, McSherry F, et al. Practical privacy: the SuLQ framework. In: Proceedings of the twenty-fourth ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems, 2005, pp.128–138. ACM, https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.126.209&rep=rep1&type=pdf
23. Li Y, Hao Z, Wen W, et al. Research on differential privacy preserving \( k \)-means clustering. Comput Sci 2013; 40(3): 287–290.
24. Dwork C. Differential privacy. In: Bugliesi M, Preneel B, Sassone V, et al. (eds) Automata, languages and programming. Berlin: Springer, 2006, pp.1–12.
25. Zhu T, Gang L, Zhou W, et al. Differentially private data publishing and analysis: a survey. IEEE Trans Knowl Data Eng 2017; 29(8): 1619–1638.
26. Langari RK, Sardar S, Mousavi S, et al. Combined fuzzy clustering and firefly algorithm for privacy preserving in social networks. Expert Syst Appl 2019; 141: 112968–112983.
27. Dou H, Chen Y, Yang Y, et al. A secure and efficient privacy-preserving data aggregation algorithm. J Ambient Intell Human Comput. Epub ahead of print 10 February 2021. DOI: 10.1007/s12652-020-02801-6.
28. Song H, Wang N, Sun J, et al. Enhanced anonymous models for microdata release based on sensitive levels partition. Comput Commun 2020; 155: 9–23.
29. Yang D, Li S, Liu Z, et al. Differentially private geospatial data publication based on grid clustering. Int J Embed Syst 2019; 11(5): 613–623.
30. Guan Z, Lv Z, Sun X, et al. A differentially private big data nonparametric Bayesian clustering algorithm in smart grid. IEEE Trans Netw Sci Eng 2020; 7(4): 2631–2641.
31. Rong WA, Benjamin CMF and Yan ZA. Heterogeneous data release for cluster analysis with differential privacy. Knowledge Based Syst 2020; 201–202: 106047.
32. Zheng X, Cai Z, Yu J, et al. Privacy-preserved data sharing towards multiple parties in industrial IoTs. IEEE J Select Areas Commun 2020; 38(5): 968–979.
33. Yang G, Ye X, Fang X, et al. Associated attribute-aware differentially private data publishing via microaggregation. IEEE Access 2020; 99: 79158–79168.
34. Li W, Zhang X, Xiaohui LI, et al. PPDP-PCAO: an efficient high-dimensional data releasing method with differential privacy protection. IEEE Access 2019; 7: 176429–176437.
35. Wang K, Chen CM, Tie Z, et al. Forward privacy preservation in IoT enabled healthcare systems. IEEE Trans Indus Inform 2021; 99: 1–9.
36. Wu TY, Lin J, Zhang Y, et al. A grid-based swarm intelligence algorithm for privacy-preserving data mining. Appl Sci 2019; 9(4): 774.
37. Chen CM, Tie Z, Wang EK, et al. Verifiable dynamic ranked search with forward privacy over encrypted cloud data. Peer Peer Netw Appl 2021; 14: 2977–2991.
38. Cao Y, Yoshikawa M, Xiao Y, et al. Differential privacy in continuous data release under temporal correlations. IEEE Trans Knowl Data Eng 2019; 31(7): 1281–1295.
39. Lu Y and Zhu M. On privacy preserving data release of linear dynamic networks. Automatica 2020; 115: 108839–108852.
40. Kanga H, Mab Y and Sia X. An enhanced algorithm for dynamic data release based on differential privacy. Procedia Comput Sci 2020; 174: 15–21.
41. Li S, Shen H, Sang Y, et al. An efficient method for privacy-preserving trajectory data publishing based on data partitioning. *China Commun* 2020; 76(7): 5276–5300.
42. Ou L, Qin Z, Liao S, et al. An optimal noise mechanism for cross-correlated IoT data releasing. *IEEE Trans Depend Secure Comput* 2020; 1: 1–12.
43. Shateri M, Messina F, Piantanida P, et al. Real-time privacy-preserving data release for smart meters. *Automatica* 2020; 11(6): 5174–5183.
44. Cai Z, Zheng X and Yu J. A differential-private framework for urban traffic flows estimation via taxi companies. *IEEE Trans Indus Inform* 2019; 15(12): 6492–6499.
45. Ma Z, Zhang T, Liu X, et al. Real-time privacy-preserving data release over vehicle trajectory. *IEEE Trans Vehi Technol* 2019; 68(8): 8091–8102.
46. Cai Z and Zheng X. A private and efficient mechanism for data uploading in smart cyber-physical systems. *IEEE Trans Netw Sci Eng* 2020; 7(2): 766–775.
47. Sweeney L. k-anonymity: a model for protecting privacy. *Int J Uncertain Fuzz Knowledge Based Syst* 2002; 10(5): 557–570.
48. Dwork C. Differential privacy: a survey of results. In: *International conference on theory and applications of models of computation*, Springer, 2008, pp.1–19, https://web.cs.ucdavis.edu/~franklin/cs289/2010/dwork_2008.pdf
49. Dwork C, McSherry F, Nissim K, et al. Calibrating noise to sensitivity in private data analysis. In: Halevi S and Rabin T (eds) *Theory of cryptography*, vol. 3876. Berlin: Springer, 2006, pp.265–284.
50. Shitov Y. Euclidean distance matrices and separations in communication complexity theory. *Discrete Comput Geometry* 2019; 61: 653–660.
51. Guendouzi W and Boukra A. A Manhattan distance-based binary bat algorithm vs. integer ant colony optimisation for intrusion detection in the audit trails. *Int J Comput Sci Eng* 2019; 18: 424–437.
52. Gao Y, Tian F and Zhanqiang W. DPk-medoids clustering algorithm based on differential privacy protection. *Comput Technol Develop* 2017; 27(10): 117–125.
53. UCI knowledge discovery archive database, http://archive.ics.uci.edu/ml/datasets.html (accessed 20 December 2020).