IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 28, 2018, accepted March 31, 2018, date of publication April 9, 2018, date of current version May 2, 2018.

Digital Object Identifier 10.1109/ACCESS.2018.2824798

DP-MCDBSCAN: Differential Privacy Preserving Multi-Core DBSCAN Clustering for Network User Data

LINA NI[1,2,3], CHAO LI[1], XIAO WANG[1], HONGLU JIANG[4], AND JIGUO YU[4],(Senior Member, IEEE)

[1]College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266590, China
[2]Shandong Province Key Laboratory of Wisdom Mine Information Technology, Shandong University of Science and Technology, Qingdao 266590, China
[3]Key Laboratory of the Ministry of Education for Embedded System and Service Computing, Tongji University, Shanghai 201804, China
[4]School of Information Science and Engineering, Qufu Normal University, Shandong 276826, China

Corresponding author: Jiguo Yu (jiguoyu@sina.com)

This work was supported by NSF of China under Grant 61672321, Grant 61771289, and Grant 61373027, in part by the Training Program of the Major Research Plan of NSF of China under Grant 91746104, in part by the National Key R & D Programs Project of China under Grant 2017YFC0804406, in part by the Project of Shandong Province Higher Educational Science and Technology Program under Grant J13LN18 and Grant J15LN19, in part by the Open Project of Tongji University Embedded System and Service Computing of Ministry of Education of China under Grant ESSCKF 2015-02.

ABSTRACT The proliferation of ubiquitous Internet and mobile devices has brought about the exponential growth of individual data in big data era. The network user data has been confronted with serious privacy concerns for extracting valuable information during the process of data mining. Differential privacy preservation is a new paradigm independent of the adversaries' prior knowledge, which protects sensitive data while maintaining certain statistical properties by adding random noise. In this paper, we put forward a differential privacy preservation multiple cores DBSCAN clustering schema based on the powerful differential privacy and DBSCAN algorithm for network user data to effectively leverage the privacy leakage issue in the process of data mining, enhancing data clustering efficaciously by adding Laplace noise. We perform extensive theoretical analysis and simulations to evaluate our schema and the results show better efficiency, accuracy, and privacy preservation effect than previous schemas.

INDEX TERMS Privacy preservation, differential privacy, MCDBSCAN clustering, data mining, Laplace noise.

I. INTRODUCTION

Internet of Things (IoT) is immersing into our daily lives and providing more comprehensive intelligent services [1]–[3]. Coupled with social networks, the explosive increasing smart devices exponentially bring about a surge of personal user data. The flourish of IoT and database techniques make the data collection no longer just the work of statistics department and the government. User data from all kinds of social network sites, shopping sites and search engines in all walks of life can be further analyzed and utilized through data mining by individuals and organizations. Unfortunately, with private or sensitive information, raw data will inevitably be in exposure and privacy leakage will be caused during this process [4]–[6]. On the other hand, in many data publishing applications which directly present the data to users in database, if data publishers do not take appropriate measures for data protection, sensitive data may be leaked. For instance, for the product information released by an enterprise or a financial annual report issued by a listed company, if the data is not carefully discriminated before publishing, it will give commercial competitors an opportunity to utilize these information. Therefore, it is a great challenge to provide privacy guarantee without significant accuracy compromise in data mining through privacy preservation techniques [7]. Privacy preservation for network user data has been received wide attentions by the society and academia in recent years [8]–[10].

Existing privacy preservation techniques mainly include data encryption, limited data publishing and data distortion, etc. Data encryption [11] adopts encryption technique

in the data mining process to hide sensitive data, which is often used in distributed environments. Limited data publishing [12] publishes data conditionally on specific circumstances in the way of publishing certain values of the data, generalizing or anonymizing the data, etc. Data distortion technique [13] distorts sensitive data while keeping some data or data attributes intact by adding noise, making exchange and randomization, blocking, etc. It may ensure that the processed data can still preserve certain statistical properties for data mining and other operations.

Differential privacy is a new paradigm of privacy tailored for statistical databases independent of the adversaries' background knowledge or computational power, which defines a rigorous attack model, reducing the risk of privacy disclosure and meanwhile ensuring the availability of data successfully [14], [15]. It is also a kind of data distortion technique. Based on differential privacy and data mining techniques, many algorithms have been presented, such as Differential Privacy Preservation K-means clustering method (DP-Kmeans), Improved Differential Privacy Preservation K-means clustering method (IDP-Kmeans) [16], Differential Privacy Preservation DBSCAN clustering method (DP-DBSCAN) [17], etc. These algorithms can achieve effective clustering via adding noise that conforms to differential privacy. However, when IDP-Kmeans faces a dataset with an unknown number of clusters and uneven density distribution, the clustering effect decreases. DP-DBSCAN is more time-consuming with less clustering for larger datasets and smaller privacy budget parameters. Therefore, it is essential to develop a novel data mining technique for privacy preservation to solve these problems.

In this paper, we focus on the differential privacy preservation in clustering analysis of network user data. By virtue of the merits of differential privacy which guarantees the data privacy independent of prior knowledge, we propose a Differential Privacy Preservation Multiple cores DBSCAN (DP-MCDBSCAN) clustering schema based on DP-DBSCAN algorithm for the network user data to effectively solve the privacy leakage in data mining. Specifically, we establish a result set of initial core points by optimizing the selection of the initial core points, and then select the desired core points from the result set for clustering.

The main contributions of this paper are summarized as follows.

1) We propose a DP-MCDBSCAN schema in clustering analysis for network user data to improve the clustering accuracy and data security. Privacy analysis demonstrates that our DP-MCDBSCAN clustering schema can not only meet the publishers' query needs but also prevent the data of publishers from being attacked.
2) We propose a multiple cores DBSCAN clustering algorithm based on differential privacy. Different from DP-DBSCAN, DP-MCDBSCAN solves the randomness and blindness of DP-DBSCAN effectively by optimizing the selection of the initial core points. The proposed algorithm also shows obvious advantages when dealing with datasets with larger scale and significant density distributions as well as smaller privacy budget parameters.
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 rest of the paper is organized as follows. Section II reviews the related work. In Section III, we give the preliminary knowledge about differential privacy and DBSCAN. Section IV proposes the DP-MCDBSCAN clustering schema. In Section V, experiments are given to verify the effectiveness of our proposed schema. Finally, we draw our conclusions and give the future work in Section VI.

II. RELATED WORK

A. DIFFERENTIAL PRIVACY PRESERVATION

Most of the existing research on differential privacy focused on theoretical properties of their proposed model to protect users' privacy.

McSherry [18] achieved a differential privacy preservation algorithm for sensitive data based on Language INtegrated Queries (LINQ), and developed the Privacy INtegrated Queries (PINQ) system which can provide some secondary development interfaces. Mohan *et al.* [19] presented GUPT (which is a Sanskrit word meaning 'Secret') which combines the data sensitivity and timeliness to gradually reduce the privacy budget. Blum *et al.* [20] proposed distributed differential privacy preservation algorithm based on interval queries and half-space queries.

Fletcher and Islam [21] proposed a differential privacy decision making forest algorithm which significantly reduces the query times and sensitivity. This approach in turn reduces the amount of noise that must be added to protect privacy, improving the availability of data.

Several works studied differential privacy in practical applications. Zhu *et al.* [22] proposed a neighbor cooperative filtering algorithm for the privacy leaking problem of K-means algorithm through differential privacy. Chen *et al.* [23] applied differential privacy to protect the transportation information in public transportation. Gotz *et al.* [24] combined differential privacy with the publishing algorithm for search log.

These algorithms effectively protect the privacy of the data. However, due to the characteristics of differential privacy, the amount of noise added will inevitably affect the availability of data. Therefore, how to balance the privacy preservation and data availability is the focus of our schema in this paper.

B. CLUSTERING ANALYSIS WITH PRIVACY PRESERVATION

Privacy preservation has become a critical concern in data mining [25]–[27]. The existing privacy preservation techniques in clustering analysis include random perturbation, data rotation, data exchange, etc.

Mukherjee *et al.* [28] proposed a data perturbation method based on Fourier transform, which guarantees that the Euclidean distance of data is invariant before and after the transformation. This approach preserves the privacy and maintains the statistical properties of data based on distance information. However, when the distribution of the dataset is unknown or non-uniform, the distance difference threshold of the algorithm is difficult to set.

Nayahi and Kavitha [29] presented an anonymous algorithm based on clustering and elastic-similar attacks as well as probabilistic reasoning attacks. This approach distributes the anonymous data in the Hadoop Distributed File System (HDFS) to achieve a better tradeoff between privacy and data availability.

Yu *et al.* [30] proposed an outlier elimination K-means algorithm based on differential privacy. Different from IDP-Kmeans [16], they utilized outlier algorithm to eliminate the interference of the outlier points, and then selected clustering core to make the core points more appropriate which reduced iteration times. However, it still can not handle the dataset with unknown cluster number.

In this paper, we propose a DP-MCDBSCAN privacy preservation clustering schema based on differential privacy, which can better solve the balance problem between privacy metrics and data availability. Under the strict privacy disclosure risk measurements, our schema achieves higher privacy standard by means of adding small amount of noise.

III. PRELIMINARIES

A. DIFFERENTIAL PRIVACY

Existing research on differential privacy mainly focused on two aspects: differentially private data publishing, and differentially private data analysis. In this paper, we mainly discuss the privacy preservation at data publishing.

During the data publishing, differential privacy disturbs the source data by adding noise and maintains some of the data and its specific attributes unchanged, so that the mined data can still maintain a certain statistical properties in some aspects [31]. The greatest advantage of differential privacy preservation is that the amount of noise added is independent of the dataset scale, and even a large dataset requires only a small amount of noise to maintain a high level of privacy preservation.

In differential privacy preservation [14], [15], the amount of noise added is related to the privacy budget parameter ε. He *et al.* [32] did a further exploration on choosing the appropriate privacy budget parameter. The method of choosing ε in our experiments is based on [32]. At the same time, since adding or deleting a piece of data record does not affect the query result, the attackers cannot judge the sensitive attributes of the unknown data record by the known one.

In this section, we introduce the basic principles of differential privacy used in this paper.

Theorem 1 [15]: Let D and D' be two neighboring datasets if they differ in at most one record. $Range(M)$ stands for the range of a random function M, $Pr[Ed]$ stands for the disclosure risk probability of event Ed, ε is the privacy budget parameter. If M provides ε-differential privacy, then for all $S_M \subseteq Range(M)$,

$$Pr[M(D) \in S_M] \leq exp(\varepsilon) \times Pr[M(D') \in S_M]. \quad (1)$$

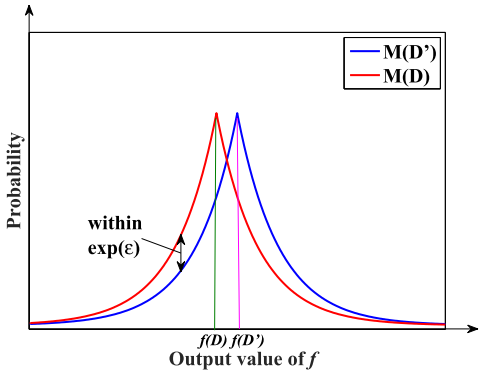

FIGURE 1. Differential privacy disclosure probability curve of dataset D and D'.

ε-differential privacy provides freedom to violate strict differential privacy for some low probability events. The disclosure risk probability of data depends on the random function M, and the choice of the random function is independent of the attackers' background knowledge.

Fig. 1 depicts the privacy disclosure risk probability curve for two neighboring datasets D and D' in the context of satisfying ε-differential privacy preservation.

Definition 1 [15] (Sensitivity): Sensitivity refers to the maximum change for the query results by deleting any records in the dataset. For a query function $f: D \to \mathcal{R}^k$, where $\mathcal{R}$ is an abstract range, k is the dimension of $\mathcal{R}$, the *sensitivity* of f is defined as

$$\Delta f = max_{D_1,D_2} ||f(D_1) - f(D_2)||_1, \quad (2)$$

where D_1 and D_2 are two neighboring datasets.

The Laplace mechanism firstly proposed by Dwork *et al.* [14] can provide a feasible method of adding noise which is the basis of differential privacy preservation.

Definition 2 [14] (Laplace Noise): Let $b = \frac{\Delta f}{\varepsilon}$, ε is the privacy budget parameter, then the *Laplace noise function* is defined as

$$Laplace(b) = exp(-\frac{|x|}{b}). \quad (3)$$

The standard deviation of the function is a symmetric exponential distribution with $\sqrt{2}b$ parameter. The probability density function of Laplace noise with the position parameter 0 and the scale parameter b is defined as

$$P(x) = \frac{exp(-\frac{|x|}{b})}{2b}. \quad (4)$$

The added noise is proportional to the value of Δf and inversely proportional to ε, that is, when Δf is small, the method performs better because less noise is added. When

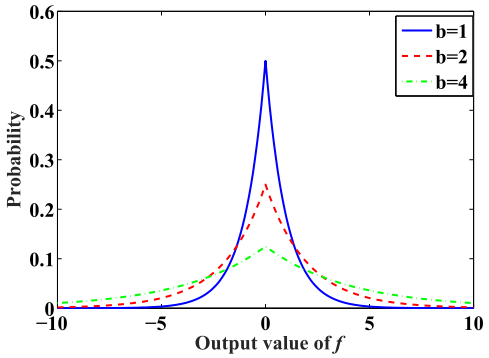

FIGURE 2. Probability density function curve of Laplace noise.

ε is reduced, the curve of $Laplace(\frac{\Delta f}{\varepsilon})$ becomes flat, which means that the amplitude of the noise is expected to become larger. When ε is fixed, the curve corresponding to the high sensitivity function f is more flat, the expected amplitude change of the nose is also large. The probability density function curve of Laplace noise is shown in Fig. 2.

Theorem 2 [14]: Let $b=\frac{\Delta f}{\varepsilon}$, f is the query function, D is the dataset, and the query result is $f(D)$. By adding Laplace noise preservation privacy to the query result, the response value of the random function M is

$$M(D) = f(D) + Laplace(b)^k, \tag{5}$$

which satisfies ε-differential privacy preservation.

B. DBSCAN CLUSTERING

DBSCAN (Density Based Spatial Clustering of Applications with Noise) is a typical clustering algorithm based on density in data mining, which can distinguish clusters with arbitrary shape. The clustering results of DBSCAN depend on the selection of parameters *Eps* (neighborhood radius) and *MinPts* (the minimum number of objects within the neighborhood radius of core object). A clustering category is obtained by dividing the samples with connected density into one class. By classifying all the groups of samples with connected density into different categories, we get all the final clustering results.

In the following, we give some concepts of DBSCAN algorithm used in this paper. For more details, please refer to [33].

Definition 3 (Eps-Neighborhood): The set of points within a given object radius *Eps* is called the *Eps-neighborhood* of the object in dataset D, denoted by $NEps(x_j)=\{x_i \in D | distance(x_i, x_j) \leq Eps\}$.

Definition 4 (Core Object): For any object $x_j \in D$, if there are at least *MinPts* objects in its *Eps-neighborhood*, that is, if $|NEps(x_j)| \geq MinPts$, then x_j is the *core object*.

Definition 5 (Directly Density-Reachable): An object x_i is said to be *directly density-reachable* from an object x_j if x_i is within the *Eps-neighborhood* of x_j, and x_j is a *core object*.

Definition 6 (Density-Reachable): x_i is *density-reachable* to x_j if there exists an object chain $p_1, p_2, \ldots, p_T$, such that $p_1 = x_i$, $p_T = x_j$ and p_{k+1} is *directly density-reachable* from p_k.

Definition 7 (Density-Connected): An object x_i is *density-connected* to object x_j with respect to *Eps* and *MinPts* if there exists a *core object* x_k such that both x_i and x_j are *directly density-reachable* from x_k with respect to *Eps* and *MinPts*.

IV. DIFFERENTIAL PRIVACY MCDBSCAN CLUSTERING SCHEMA

A. SYSTEM ARCHITECTURE

We consider a scenario of clustering job-seeking information. Here we use a information set including 10 million high-profile resumes as a complete database, denoted as $D = \{D_1, D_2, \ldots, D_m\}$. A resume includes multiple attributes such as education, salary, position, company size, etc., where part of the resumes contain full fields and part of the resumes exist blank items. In some scenarios, analysts want to learn, encode and test the data, mine the direction and law of the position path to form an algorithm model, and then predict the blank information in the dataset.

Analysts can sort out and mine some of these attributes separately in mining data. Each attribute can be used as a dimension, hence, this is a multi-dimensional clustering model. For example, analysts can only mine two attributes, such as education and salary. First, the data is preprocessed and converted into the data in interval [0, 1] by normalization operation. Then our DP-MCDBSCAN algorithm can be used to carry out the mining process. Each resume is an object p in dataset D. Through pre-setting the *Eps-neighborhood*, the data density near each p can be calculated, where there are only the two dimensions of education and salary. The dataset is divided into different clusters according to the degree of density aggregation, and the results can be obtained after further sorting. The vacant salary item can be predicted according to education degree.

There are some sensitive attributes (such as the job seekers' name, age, contact, etc.) in those resumes which will inevitably leak in the mining process if not protected well. To solve the problem, we introduce differential privacy preservation to DBSCAN clustering algorithm, and set the corresponding privacy budget parameter ε according to the required preservation level. After adding a certain amount of noise, analysts are unable to mine the sensitive attributes through the known information. One of the great advantages is that the amount of noise added has nothing to do with the size of the dataset. Even for 100,000 copies of the resumes, a higher level of preservation can also be achieved through a small amount of noise.

By summing up the above application scenarios, we can abstract the general scenario in which our schema applies. The system architecture of DP-MCDBSCAN schema is shown in Fig. 3. In the following, we depict the principle of our schema.

As can be seen from Fig. 3, by means of computers, mobile phones, pad and other intelligent terminals, users

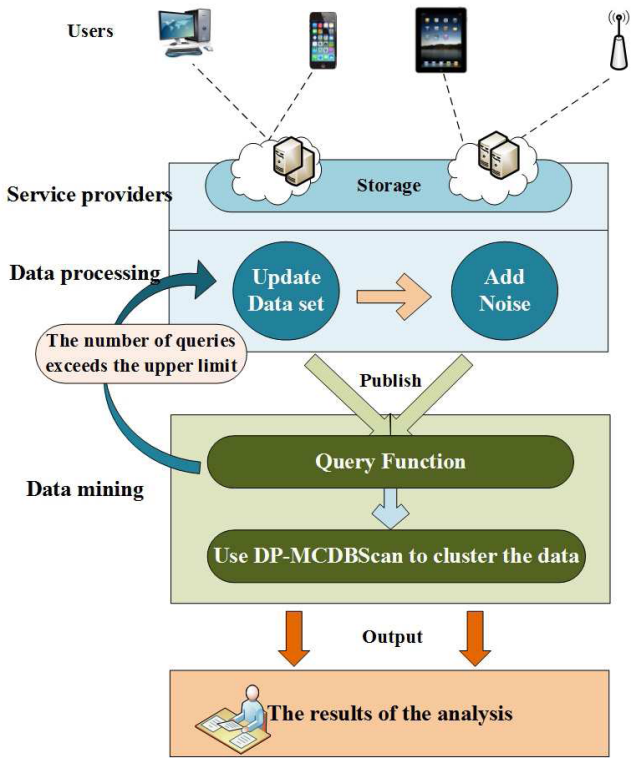

FIGURE 3. System architecture of DP-MCDBSCAN schema.

using a variety of services will produce massive data which are stored in the service provider servers (SPs). In order to find the rules from the massive data, predict the user's behavior preferences and do some statistics, SPs regularly publish and update the datasets. Meanwhile, SPs add Laplace noise to protect the users' privacy before publishing the data. Faced with publishing data, the required results are obtained through the query functions and then clustered employing our DP-MCDBSCAN algorithm. For the sake of continuous validity of privacy preservation, the users need to wait for the publishers to update the datasets before the number of queries reaches the upper limit. Finally, the clustering results are analyzed to obtain the rules of statistics and predict the behavior preference of users.

B. DP-MCDBSCAN CLUSTERING

1) DP-DBSCAN ALGORITHM

Our DP-MCDBSCAN algorithm is proposed on the basis of DP-DBSCAN algorithm [17]. In the following, we give a briefly introduction of DP-DBSCAN algorithm.

In DP-DBSCAN algorithm, a cluster can be uniquely identified by any of the core objects. That is, for any data object p that satisfies the core object condition, all the sets of data objects O_i in dataset D *density-reachable* from p constitute a complete clustering C and $p \in C$.

DP-DBSCAN clustering algorithm calculates the Euclidean distance between two points in the *Eps-neighborhood* from the core objects, and determines whether the two points are *density-reachable* to form a new cluster. For a class of queries, if an exact distance between two points is published in the cluster analysis, the attackers can infer the specific information between the two points from the known object radius *Eps*, hence the sensitive attributes have the revealing possibility.

Based on the privacy preservation data analysis [34], DP-DBSCAN adds noise to each dimension of the *direct density-reachable* points in the dataset by differential privacy technique so that the published data can conform to the privacy budget requirement, thereafter the privacy of the data is protected during clustering.

2) DP-MCDBSCAN ALGORITHM

Since DP-DBSCAN publishes the approximation of data points density, the attackers cannot deduce the sensitive properties of the data points even if they grasp some information through the knowledge background. However, when the privacy budget parameter ε is small (i.e., the added noise is too large), the accuracy of DP-DBSCAN clustering algorithm will decrease. Moreover, when the data size is large and the density is non-uniform, the clustering efficiency will also decrease.

In order to solve the drawbacks of DP-DBSCAN where the initial core object is randomly selected, we propose a DP-MCDBSCAN (Differential Privacy Preservation Multi-core DBSCAN Clustering) algorithm which determines multiple core objects as the initial object to cluster through the furthest distance selection method. Our algorithm ensures that the initial cluster centers are dispersed as far as possible so that the initial core objects selected are not in the same cluster, reducing the influence of the initial core objects selection on the clustering result.

Our DP-MCDBSCAN algorithm comprises the following seven steps.

Step 1: Let $X=\{x_1, x_2, \ldots, x_n\}$ and $Y=\{y_1, y_2, \ldots, y_n\}$ be two points *directly density-reachable* in dataset D with n-dimensional space $[0, 1]^n$.

In the dataset with n-dimensional space $[0, 1]^n$, the point distance between X and Y is

$$dis(X, Y) = \sum_{i=1}^{n} (x_i, y_i)^2.$$

Let $b = \frac{\Delta f}{\varepsilon}$. Add random noise to each dimension to get

$$dis'(X, Y) = \sum_{i=1}^{n} (x_i, y_i)^2 + Laplace(b),$$

where

$$Laplace(b) = exp(-\frac{|x|}{b}) = exp(-\frac{x \times \varepsilon}{\Delta f}).$$

Repeat the above process until all points are already contained in any cluster or are marked as "noise", the algorithm ends.

Step 2: Select the two distance-farthest core points P and Q from N samples of dataset.

Step 3: $d(P, Q)$ is the distance of the encrypted points P and Q, make the following decision:

1) If $d(P, Q) > Eps$, continue.
2) If $d(P, Q) \leq Eps$, turn to *Step* 6.

Step 4: Determine whether the core object comes up through above steps. If it is the core object, it will be added to the core objects set *Core*(); otherwise, this point is removed, and the dataset sample point is changed into $N \leftarrow N-1$.

Step 5: According to the number of core objects in *Core*() set, denoted by *count*(*Core*()), do the following:

1) If *count*(*Core*())=0, go to *Step* 2 and find the initial objects again.
2) If *count*(*Core*())=1, find the furthest point from the first point in the remaining objects, and then go to *Step* 3.
3) If *count*(*Core*())=2, find the sample point P_3 satisfying the following formula in the remaining sample points:

$$d(P_1, P_3) * d(P_2, P_3) \geq d(P_1, P_i) * d(P_2, P_i)$$

where P_i is any point in the remaining sample points except P_3.
Then go to *Step* 3 to determine the relationship between P_3 and all points in *Core*(). If its distance from any point in *Core*() is less than or equal to *Eps*, go to *Step* 6 directly.
4) If *count*(*Core*())>2, find the point P_j satisfying the following formula in the remaining sample points:

$$d(P_1, P_j) * d(P_2, P_j) * \ldots * d(P_{count(Core())}, P_j) \geq d(P_1, P_i) * d(P_2, P_i) * \ldots * d(P_{count(Core())}, P_i)$$

Then go to *Step* 3 to determine the relationship between d_j and all points in *Core*(). If the distance from any point in *Core*() is less than or equal to *Eps*, go to *Step* 6 directly.

Step 6: Find all the *directly density-reachable* points in the *Eps-neighborhood* of each core object in *Core*().

Step 7: Find the maximum density connected set by the *directly density-reachable* points of the object in *Core*().

The detailed pseudocode of our DP-MCDBSCAN is elaborated in Algorithm 1.

3) CORRECTNESS OF THE ALGORITHM

In the following, we give the correctness proof of DP-MCDBSCAN.

Theorem 3: DP-MCDBSCAN algorithm satisfies ε-differential privacy preservation.

Proof: Sensitivity refers to the biggest change to the query result caused by deleting any record in the dataset, which is the nature of the query function f itself, regardless of the size of the dataset. The sensitivity Δf of most query function is smaller. Specially, sensitivity $\Delta f = 1$ for the counting query function.

Let D_1 and D_2 be the adjacent datasets with only one record difference.

Algorithm 1 DP-MCDBSCAN

Input: D={$P_1, P_2, \ldots, P_n$}: a dataset of n points in d dimensions, *Eps*: neighborhood radius, *MinPts*: the minimum number of points in the neighborhood radius of the core points, ε: privacy budget parameter.
Output: n clusters $C = \{C_1, \ldots C_n\}$.

Add noise to the distance of data points in D, calculate the distance of the data points after adding noise:

$$dis' = \sum_{i=1}^{n} (x_i, y_i)^2 + Laplace(a),$$

$$Laplace(a) = exp(-\tfrac{x \times \varepsilon}{\Delta f})$$

MCDBCAN(D, *Eps*, *MinPts*) {
  $C = 0$
  for each point P in dataset D **do**
    $NeighborPts = regionQuery(P, Eps)$
  end for
  Add all core points to *Dcore* set
  Select the farthest core points p_1, q_1 from *Dcore*
  Add p_1, q_1 to *Core*() set
  while *Dcore*! = *null* **do**
    for each point P in *Dcore* **do**
      if $dist(P, P') < Eps$ **then**
        Delete P from *Dcore*
      end if // P' is any point in *Dcore*
    end for
    Select the maximum point satisfied

$$\sum_{i=1}^{count(core)} dist(P, P')$$

    to join *Core*() set
  end while
  for each point P in core **do**
    C=next cluster
    expandCluster(P, *NeighborPts*, C, *Eps*, *MinPts*)
  end for
  for each point P in dataset D **do**
    Mark P as "NOISE"
  end for
}
expandCluster(P, *NeighborPts*, C, *Eps*, *MinPts*) {
  Add P to cluster C
  for each point P' in *NeighborPts* **do**
    if P' is not visited **then**
      Mark P' as "visited"
    end if
    if P' is not yet a member of any cluster **then**
      Add P' to cluster C
    end if
  end for
}
regionQuery(P, *Eps*) {
  Return all points within P's *Eps*-neighborhood (including P)
}

When adding or deleting a record in n-dimensional space $[0, 1]^n$, the sensitivity of each dimension $\Delta f = 1$.

The sensitivity of the whole query sequence $\Delta f = n$.

Let $Par(D_1)$ and $Par(D_2)$ denote the clustering results after adding the noise of D_1 and D_2 respectively, and S denotes any kind of clustering.

Then, according to Theorem 1 and Theorem 2, we have

$$\frac{Pr[Par(D_1) = S]}{Pr[Par(D_2) = S]} \leq exp(\varepsilon).$$

Thus, we proved that DP-MCDBSCAN algorithm satisfies ε-differential privacy preservation.

C. PRIVACY ANALYSIS

In this section, we elaborate the privacy analysis of our DP-MCDBSCAN schema. We assume that the analysts are semi-trusted. The data publishers should not only be met their query needs but also prevent users' privacy from being attacked.

1) PUBLISHERS

Data publishers need to publish data frequently without knowing the analysts' background. In order to prevent analysts from finding individual differences by differentiating data published at different time points and accessing to individual data further, the noise needs to be added. Thus, each published data does not affect the query results due to the existence of a record. The Laplace noise added to DP-MCDBSCAN is the effective noise calculated from the sensitivity of the query function.

2) ANALYSTS

The analysts send query through the query function and receive the result. Usually, they do not contact the users or the data publishers. They only employ the query function to get the statistical rules in the data. After the processing of DP-MCDBSCAN, they can only get the noised data, and can't identify individual differences by the knowledge background. Even though they have mastered part of the user's information by linking to other databases, no other user data can be inferred because they do not know the amount of noise added.

3) DIFFERENTIAL PRIVACY

In our schema, Laplace noise is added to every query. No one knows the amount of noise added, and therefore they cannot infer the personal data by removing the noise. Hence, differential privacy can perfectly protect the security of a single query. The privacy budget parameter ε is used to balance the privacy preservation level and data accuracy. A smaller privacy budget parameter ε means higher preservation level and lower data accuracy.

However, different privacy has its own limitations. As the number of queries increases, the level of privacy preservation will be reduced. Although the answer to a single query is accordance with ε-differential privacy, we can't implement it when many queries are answered unless they are manipulated on different disjoint subsets of dataset. In the setup of our schema, the queries are made on a random sample of the original data which is constantly changing. If the data subset of the query is somewhat non-trivial, they will overlap each other and interconnected.

In order to ensure the privacy of both multiple queries and single query, we have chosen to set the upper limit K of the queries when the original data set is not large enough.

For the same data subset, when the query reaches the upper limit K, the server will suspend the service and wait until all data is updated. When the update is complete, the new dataset will be a completely different set of previous dataset. At this moment, the query counter is reset and the service is opened again.

V. EXPERIMENTS

In this section, we implement DP-MCDBSCAN and evaluate its performance via extensive experiments.

A. EXPERIMENT SETUP

In order to evaluate the performance of our DP-MCDBSCAN algorithm, we conduct the experiments on four datasets of UCI [35] with different database properties and scale: *Wine*, *Haberman*, *Waveform Database* and *MAGIC*. The details of the datasets are shown in Table I, which includes their alias, data type, number of attributes and records.

TABLE 1. Experimental Dataset Characteristics

DataSet	Alias	Type	Number of Attributes	Number of Records
Wine	D_1	Real	13	178
Haberman	D_2	Real	4	306
Waveform	D_3	Real	40	5000
MAGIC	D_4	Real	11	19020

Firstly, the datasets are preprocessed to be normalization and the values of each attribute are controlled within the same interval. In order to minimize the impact of parameters *Eps* and *MinPts*, the preprocessing takes 1/25 of the dataset scale as the value of *MinPts* and *Eps* is adjusted gradually with a gradient of 0.1. The optimal values of *Eps* and *MinPts* of each dataset are determined by observing the clustering effect. The privacy level is controlled under the determined *Eps* and *MinPts* values, thus the clustering validity of the algorithm under different privacy levels is evaluated.

In our experiments, the average results are reported by running each test dataset one hundred times independently, and all the experiments are run on Intel (R) Core (TM) i7-4700MQ CPU@3.4GHz with 8GB memory in operating system Win10 X64 Ultimate.

B. EVALUATION METRICS

1) F-MEASURE INDEX

F-measure [36] is one of the commonly used evaluation indexes of clustering results, which can measure the availability of clustering results. If the clustering results of two clustering algorithms are calculated by F-measure, the F value will be proportional to the similarity of the results.

The formula for the F-measure evaluation index is as follows:

$$P = Precision(C_i, D_j) = \frac{n_{ij}}{|D_j|} \tag{6}$$

$$R = Recall(C_i, D_j) = \frac{n_{ij}}{|C_i|} \tag{7}$$

$$F_i = \frac{2 * P * R}{P + R} \tag{8}$$

where C_i and D_j are the clustering results of two clustering algorithms, n_{ij} denotes the number of objects at the intersection of cluster C_i of D_j, P is the precision, and R is the recall rate.

A higher F-measure value means the algorithm has more clustering availability. In this paper, the F-measure value is used to evaluate the clustering results of DP-MCDBSCAN and DP-DBSCAN algorithms. We run the two algorithms respectively for clustering by setting up different privacy budget parameter ε. The clustering results are compared with the ones provided by the original dataset to judge the availability of final results.

2) CALINSKI-HARABASZ INDEX

Calinski-Harabasz [37] is also an evaluation index (hereinafter referred to as *CH* index for short) to evaluate the validity of clustering. *CH* index is the ratio of the closeness of the class described by the deviation matrix within the class and the separation degree of the classes described by the deviation matrix between the classes. *CH* index is defined as follows:

$$CH(K) = \frac{trB(k)/(k-1)}{trW(k)/(n-k)} \tag{9}$$

where n represents the number of clustering, k represents the number of current classes, $trB(k)$ represents the trace of the deviation matrix between the classes, and $trW(k)$ represents the trace of the deviation matrix within the class.

$trB(k)$ refers to the sum of squares of the distance between the center points of each cluster and the center point of the dataset, which is used to measure the separation degree of the dataset. $trW(k)$ refers to the sum of squares between the points in the class and the center of the cluster, which is used to measure the closeness of the cluster. *CH* is the ratio of those. The larger the *CH* value, the more compact the cluster in the class is, the more dispersed the cluster between the class is, the better the clustering validity is.

C. ANALYSIS OF EXPERIMENTAL RESULTS

We perform differential privacy preservation clustering by running the two algorithms on dataset D_2, and the clustering results are shown in Figs. 4 and 5.

Fig. 4 is the results of DP-DBSCAN clustering and Fig. 5 is the results of the DP-MCDBSCAN clustering. In both figures, the red hollow circles represent the noise points, and different colors represent different clusters. The cross symbols indicate the core points in the clusters. The solid dots indicate the boundary points of each cluster.

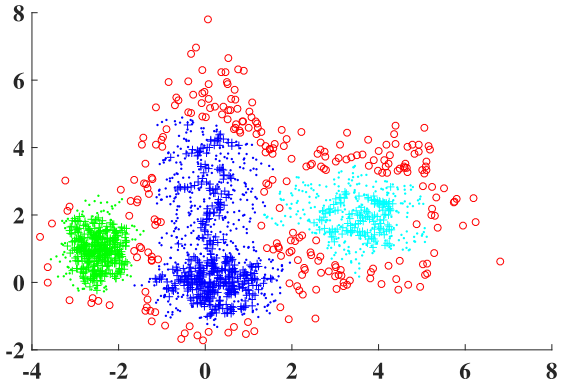

FIGURE 4. Clustering results of DP-DBSCAN algorithm.

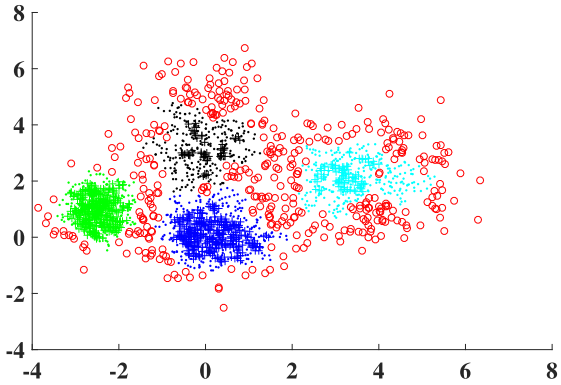

FIGURE 5. Clustering results of DP-MCDBSCAN algorithm.

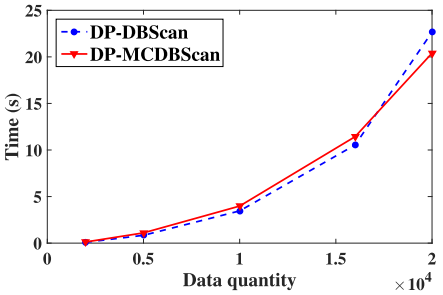

FIGURE 6. Run time comparison of two algorithms on D_4.

It can be seen from the figures that both the two methods can accurately distinguish the core points, boundary points and noise points of each cluster. However, because the way of the initial core point selection of the two algorithms is different, the effect of the cluster classification formed by DP-MCDBSCAN is different from that of DP-DBSCAN. Since DP-MCDBSCAN is clustered initially via multi-core clustering, the number of the cluster classification formed by DP-MCDBSCAN is more than that of DP-DBSCAN, and the classification is more detailed. DP-MCDBSCAN can still accurately distinguish the noise points for the datasets with non-uniform density. Thus, it is not easy to miss some cases where the distribution is similar but different clusters are clustering.

Fig. 6 shows the comparison of two algorithms in terms of run efficiency using the subset of dataset D_4 with different

data volumes. The two algorithms is executed numerous times in D_4 respectively. Comparing the time efficiency of the two algorithms, it can be seen that the time of DP-MCDBSCAN is slightly higher than that of the original DP-DBSCAN with smaller data. The main reason is that the basic time in the initial determination of multiple core points is relatively longer than the total time spent.

However, when the dataset reaches a certain scale, here the data quantity is $1.67 * 10^4$, the run time superiority of DP-MCDBSCAN appears, which is that after the initial core points is determined, the following only need to determine the remaining points of each cluster. In contrast, because of the data volume of DP-DBSCAN as well as the clusters increased, a large number of the following existing iterative computing is used to find new core points, much time is spent. We can conclude from Fig. 6 that our DP-MCDBSCAN algorithm has more superiorities in the face of the dataset with larger scale or the dataset with more clusters.

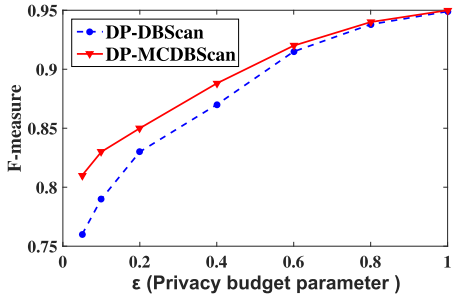

FIGURE 7. *F*-measure comparison of two algorithms.

Next, we consider the impact of F-measure on the privacy budget parameter $\varepsilon \in \{0.1, 0.2, \ldots, 0.5, \ldots, 1\}$, ranging from strict to loose privacy requirements. Fig. 7 shows the comparison of the F-measure values for the clustering results of two algorithms on dataset D_3 under different ε. It can be seen that the accuracy of the DP-MCDBSCAN is slightly higher than that of the DP-DBSCAN. When ε is large enough, the clustering of both algorithms is accurate and effective. Faced with smaller ε, DP-MCDBSCAN can better deal with the dataset.

Meanwhile, differential privacy has the characteristic that the added noise has nothing to do with the dataset scale. Therefore, DP-MCDBSCAN has the greater superiority in handling large scale dataset. The larger the scale of the dataset, the stronger the noise immunity and the better performance the DP-MCDBSCAN will have.

In our last experiment, we select D_1, D_2 and D_3 as our test datasets. We compare our algorithm with DP-DBSCAN through the *Calinski-Harabasz* (*CH*) index to further evaluate the clustering validity. We perform privacy-preserving clustering and analyze on D_1, D_2 and D_3 via executing the two algorithms multiple times. We utilize the mean value of *CH* and plot the *CH* ratio curve of the two algorithms. The closer to 1 the *CH* ratio is, the closer the clustering validity of the two algorithms is. The experimental results are shown in Figs. 8, 9 and 10.

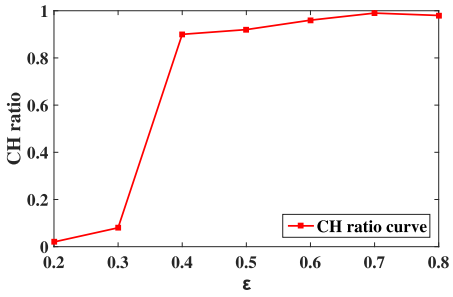

FIGURE 8. *CH* ratio on D_1.

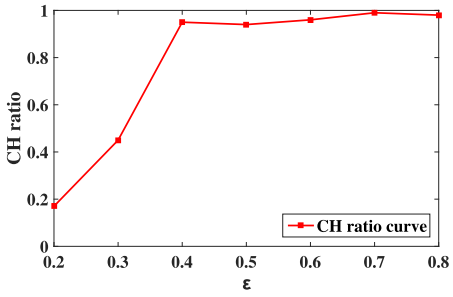

FIGURE 9. *CH* ratio on D_2.

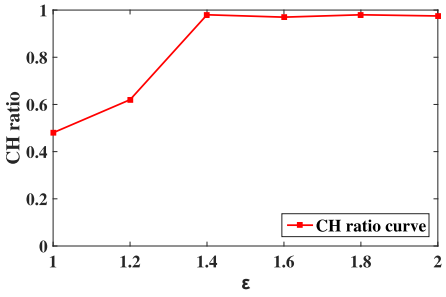

FIGURE 10. *CH* ratio on D3.

The results of the figures show that DP-MCDBSCAN can achieve a good effect on privacy preservation by adding a small amount of noise. We also find that it ensures that DP-MCDBSCAN has the similar clustering validity as the traditional DP-DBSCAN clustering algorithm. Especially, it indicates that the level of privacy preservation depends on the value of ε. The privacy preservation level can be controlled by the value of ε. The smaller the ε, the more noise added and the higher the privacy preservation level.

By comparing the results of the three figures, we conclude that under the same level of privacy preservation (i.e., the same ε), DP-MCDBSCAN has the following features: the clustering validity for smaller dataset is higher than that for larger dataset, and the clustering validity for lower dimensional dataset is higher than that for high-dimensional dataset.

VI. CONCLUSION

Ensuring privacy security of network user data in data mining is an important and challenging problem. In this paper,

we focus on the privacy preservation in clustering analysis of network user data. We have proposed a DP-MCDBSCAN schema and the corresponding algorithm. Different from the previous works, we adopt the multiple cores selection method at the farthest distance on the clustering result to solve the randomness and blindness of DP-DBSCAN effectively. Simulation results show that our algorithm reduces the effect of clustering when the added noise is too large and the time effectiveness of the results is enhanced. Due to the characteristics of differential privacy, the amount of noise added is independent of the size of dataset, so the clustering effect of the larger scale dataset is more accurate under the same privacy budget parameter.

Our future work will focus on reducing the influence of input parameters on clustering results and balancing the influence between adding noise and the accuracy of clustering.

REFERENCES

[1] T. Song, R. Li, B. Mei, J. Yu, X. Xing, and X. Cheng, "A privacy preserving communication protocol for IoT applications in smart homes," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1844–1852, Dec. 2017.

[2] R. Li, T. Song, N. Capurso, J. Yu, J. Couture, and X. Cheng, "IoT applications on secure smart shopping system," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1945–1954, Dec. 2017.

[3] T. Song, N. Capurso, X. Cheng, J. Yu, B. Chen, and W. Zhao, "Enhancing GPS with lane-level navigation to facilitate highway driving," *IEEE Trans. Veh. Technol.*, vol. 66, no. 6, pp. 4579–4591, Jan. 2017.

[4] X. Zheng, G. Luo, and Z. Cai, "A fair mechanism for private data publication in online social networks," *IEEE Trans. Netw. Sci. Eng.*, to be published, doi: 10.1109/TNSE.2018.2679483.

[5] Y. Liang *et al.*, "Location privacy leakage through sensory data," *Secur. Commu. Netw.*, vol. 2017, Aug. 2017, Art. no. 7576307.

[6] N. Capurso, T. Song, W. Cheng, J. Yu, and X. Cheng, "An Android-based mechanism for energy efficient localization depending on indoor/outdoor context," *IEEE Internet Things J.*, vol. 4, no. 2, pp. 299–307, Apr. 2017.

[7] X. Zheng, Z. Cai, J. Yu, C. Wang, and Y. Li, "Follow but no track: Privacy preserved profile publishing in cyber-physical social systems," *IEEE Internet Things J.*, vol. 4, no. 6, pp. 1868–1878, Dec. 2017.

[8] H. N. Chua, A. Herbland, Y. Chang, and S. F. Wong, "Compliance to personal data protection principles: A study of how organizations frame privacy policy notices," *Telematics Inform.*, vol. 34, no. 4, pp. 157–170, Jul. 2017.

[9] C. Hu, W. Li, X. Cheng, J. Yu, S. Wang, and R. Bie, "A secure and verifiable access control scheme for big data storage in clouds," *IEEE Trans. Big Data*, to be published, doi: 10.1109/TBDATA.2016.2621106.

[10] Z. He, Z. Cai, and J. Yu, "Latent-data privacy preserving with customized data utility for social network data," *IEEE Trans. Veh. Technol.*, vol. 67, no. 1, pp. 665–673, Jan. 2018.

[11] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Y. Zhu, "Tools for privacy preserving distributed data mining," *ACM SIGKDD Explorations Newslett.*, vol. 4, no. 2, pp. 28–34, Dec. 2002.

[12] L. Sweeney, "Achieving k-anonymity privacy protection using generalization and suppression," *Int. J. Uncertainty, Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 571–588, Oct. 2002.

[13] Y. Saygin, V. S. Verykios, and A. K. Elmagarmid, *Privacy Preserving Association Rule Mining*. New York, NY, USA: Springer, 2002.

[14] C. Dwork, F. McSherry, K. Nissim, and A. D. Smith, "Calibrating noise to sensitivity in private data analysis," in *Proc. 3rd Theory Cryptograph. Conf. (TCC)*, New York, NY, USA, Mar. 2006, pp. 265–284.

[15] C. Dwork, "Differential privacy: A survey of results," in *Proc. Theory Appl. Models Comput. Int. Conf.*, Apr. 2008, pp. 1–19.

[16] Y. Li *et al.*, "Research on differential privacy preserving K-means clustering," *Comput. Sci.*, vol. 40, no. 3, pp. 287–290, Mar. 2013.

[17] W. Wu and H. Huang, "A DP-DBScan clustering algorithm based on differential privacy preserving," *Comput. Eng. Sci.*, vol. 37, no. 4, pp. 830–834, 2015.

[18] F. D. McSherry, "Privacy integrated queries: An extensible platform for privacy-preserving data analysis," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (MD)*, Jun. 2009, pp. 19–30.

[19] P. Mohan, A. Thakurta, E. Shi, D. Culler, and D. Song, "GUPT: Privacy preserving data analysis made easy," in *Proc. ACM SIGMOD Int. Conf. Manage. Data (MD)*, May 2012, pp. 349–360.

[20] A. Blum, K. Ligett, and A. Roth, "A learning theory approach to non-interactive database privacy," *J. JACM*, vol. 60, no. 2, pp. 12:1–12:25, Apr. 2013.

[21] S. Fletcher and M. Z. Islam, "Differentially private random decision forests using smooth sensitivity," *Expert Syst. Appl.*, vol. 78, pp. 16–31, Jul. 2017.

[22] T. Zhu, G. Li, Y. Ren, W. Zhou, and P. Xiong, "Differential privacy for neighborhood-based collaborative filtering," in *Proc. IEEE/ACM Inter. Conf. Adv. Social Netw. Anal. Mining (ASNAM)*, Aug. 2013, pp. 752–759.

[23] R. Chen, B. C. M. Fung, B. C. Desai, and N. M. Sossou, "Differentially private transit data publication: A case study on the montreal transportation system," in *Proc. 18th ACM SIGKDD Inter. Conf. Knowl. Discovery Data Mining (KDDM)*, Aug. 2012, pp. 213–221.

[24] M. Gotz, A. Machanavajjhala, G. Wang, X. Xiao, and J. Gehrke, "Publishing search logs—A comparative study of privacy guarantees," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 3, pp. 520–532, Mar. 2012.

[25] K. Xing, C. Hu, J. Yu, X. Cheng, and F. Zhang, "Mutual privacy preserving k-means clustering in social participatory sensing," *IEEE Trans. Ind. Informat.*, vol. 13, no. 4, pp. 2066–2076, Aug. 2017.

[26] Z. He *et al.*, "Cost-efficient strategies for restraining rumor spreading in mobile social networks," *IEEE Trans. Veh. Technol.*, vol. 66, no. 3, pp. 2789–2800, Jun. 2017.

[27] Z. Cai, Z. He, X. Guan, and Y. Li, "Collective data-sanitization for preventing sensitive information inference attacks in social networks," *IEEE Trans. Depend. Sec. Comput.*, to be published, doi: 10.1109/TDSC.2016.2613521.

[28] S. Mukherjee, Z. Chen, and A. Gangopadhyay, "A privacy-preserving technique for Euclidean distance-based mining algorithms using Fourier-related transforms," *Int. J. Very Large Data Bases*, vol. 15, no. 4, pp. 293–315, Nov. 2006.

[29] J. J. V. Nayahi and V. Kavitha, "Privacy and utility preserving data clustering for data anonymization and distribution on hadoop," *Future Generat. Comput. Syst.*, vol. 74, pp. 393–408, Sep. 2017.

[30] Q. Yu, Y. Luo, C. Chen, and X. Ding, "Outlier-eliminated k-means clustering algorithm based on differential privacy preservation," *Appl. Intell.*, vol. 45, no. 4, pp. 1179–1191, Dec. 2016.

[31] T. Zhu *et al.*, "Differentially private data publishing and analysis: A survey," *IEEE Trans. Knowl. Data Eng.*, vol. 29, no. 8, pp. 1619–1638, Aug. 2017.

[32] X. He *et al.*, "Study on choosing the parameter ε in differential privacy," *J. Commu.*, vol. 36, no. 12, pp. 124–130, 2015.

[33] M. Ester *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," *Proc. 2nd Int. Conf. Knowl. Disco. Data Mining*, 1996, pp. 226–231.

[34] C. Dwork, "A firm foundation for private data analysis," *Commu. ACM*, vol. 54, no. 1, pp. 86–95, Jan. 2011.

[35] *UCI Datasets*. Accessed: Dec. 27, 2017. [Online]. Available: http://archive.ics.uci.edu/ml/datasets.html

[36] D. Powers, "Evaluation: From precision, recall and F-factor to ROC, informedness, markedness and correlation," *J. Mach. Learn. Tech.*, vol. 2, p. 2229, Jan. 2011.

[37] T. Caliński, H. Ja, and J. Harabasz, "A dendrite method for cluster analysis," *Commu. Statist.*, vol. 3, no. 1, pp. 1–27, Jan. 1974.

LINA NI received the Ph.D. degree in computer software and theory from Tongji University, Shanghai, China, in 2009. She is currently an Associate Professor with the College of Computer Science and Engineering, Shandong University of Science and Technology, Shandong. She is also the Committee Member of Professional Committee of Network Information Service of China Automation Federation. Her current areas of research are privacy preservation, cloud computing, petrinet, distributed algorithms, and intelligent computing. She is a member of the ACM, and a senior member of the China Computer Federation.

CHAO LI is currently pursuing the M.S. degree with the College of Computer Science and Engineering, Shandong University of Science and Technology. His main research interests include privacy preservation, cloud computing, and distributed algorithms. He is a member of the China Computer Federation.

XIAO WANG is currently pursuing the M.S. degree with the College of Computer Science and Engineering, Shandong University of Science and Technology. Her main research interests include privacy preservation, cloud computing, and machine learning.

HONGLU JIANG received the B.S. and M.S. degrees in computer science from Qufu Normal University, in 2009 and 2012, respectively, where she is currently pursuing the Ph.D. degree with the School of Information Science and Engineering. Her research interests include wireless networks, distributed computing, and privacy preserving.

JIGUO YU (SM'17) received the Ph.D. degree from the School of Mathematics, Shandong University, in 2004. He was a Full Professor with the School of Computer Science, Qufu Normal University, Shandong, China, in 2007, where he is currently a Full Professor with the School of Information Science and Engineering. His main research interests include privacy-aware computing, wireless networking, distributed algorithms, peer-to-peer computing, and graph theory. He is a member of the ACM and the China Computer Federation.