Finding Quasi-identifiers for K-Anonymity Model by the Set of Cut-vertex

Yan Yan, Wanjun Wang, Xiaohong Hao, Lianxiu Zhang

Abstract—**The rapid development of data publishing and information access technology bring a growing number of problems in privacy leakage. In order to avoid linking attacks happened between attributes, K-anonymity model was proposed and become the most widely used in privacy preserving data publishing. Identification of quasi-identifiers (QIs) is one of the primary problems which will directly affect the effectiveness of K-anonymity method. However, most of the existing methods ignored this problem or just choose QIs empirically. These will greatly reduce the validity of K-anonymity method as well as the utility of anonymous data. In this paper, we study the problem of finding QIs for privacy preserving data publishing method based on K-anonymity model. Firstly, we analyze the roles of QIs from the view of independence of sets, and define it as a collection of attributes that can separate sensitive attributes from the other non-sensitive attributes. Then, we propose a construction method for attribute graph based on relationship matrix, which can represent potential connectivity of publishing data, published data and external knowledge. Finally, we put forward an identification algorithm for QIs based on the concept of cut-vertex, which is aiming to find the necessary and minimum QIs. The proposed algorithm is useful to avoid inconvenience and inaccuracy caused by artificial partition of QIs, and can be applied in data publishing situations with multiple sensitive attributes after some extension. Experiments and analysis show that the proposed identification algorithm has better partition ability and lower computational complexity. Therefore, it has good practical value in the application environment of publishing of big data.**

Index Terms—**privacy preserving data publishing; k-anonymity; quasi-identifiers; attributes graph; cut-vertex**

I. Introduction

The rapid development of cloud computing, Internet of things and big data technology bring about great convenience for accessing of information. Companies, governments and organizations are publishing and sharing more and more micro-data for the purpose of research or business. However, improper collection, analysis and publication of data will lead to privacy leakage. In recent years, there have been many global information security incidents caused by data leakage. On October 2015, more than 4 million users' data have been leaked by the broadband service provider TalkTalk in UK. Including user's name, address, date of birth, phone number, e-mail and other privacy information. In 2016, the largest cable TV company in the United States — Time Warner said that about 32 million users' e-mail and password information have been stolen by hacker. In the same year, great data leakage event has broken out in Turkey. Nearly 50 million citizens' personal information was hacked for people to download. In 2017, more than 220 million users' information has been revealed (including name, e-mail address, phone number, home address, family coordinates and social profile links) in India, because the API endpoint of McDelivery Company has not been protected. In the same year, the database of business services giant Dun & Bradstreet was leaked, which included about 33.8 million e-mail addresses and other contact information from thousands of employees and government departments in the United States. The world's largest loopholes response platform — Butian published information that more than 30 provinces in China have high-risk loopholes in their social security, household registration, centers for disease control and many other systems. Only the data involved by social security vulnerabilities reached the amount of 5279.4 million, including identity cards, social security information, finance, income, housing and other sensitive information. Security incidents mentioned above have already sounded the alarm of information security. How to manage the privacy of data and prevent leakage of personal information under the environment of big data has become an issue of common concern for the whole world [1-6].

Generally speaking, privacy can be defined as some information that is not willing to be known by outside world. For individuals, privacy is related to their personal sensitive information, for example, salaries, medical records, investment situations, account and password of transactions, financial information, etc. Therefore, in most of the data publishing methods, attributes are usually partitioned into explicit identifiers, quasi-identifiers (QIs), and sensitive attributes. Explicit identifiers refer to attributes that can explicitly and uniquely identify an individual, such as Name or ID Number. QIs refer to a subset of attributes. None particular tuple can be uniquely identified when using the attribute alone, but when these kinds of attributes are taken together, it can potentially identify an individual. For example, the combination of Gender, Date of birth and Zip code can determine about 87% of the population in United States. Therefore, incautious publication of QIs will lead to privacy leakage. Traditional privacy preserving data publishing methods delete identifier attributes to protect personal privacy, but it still cannot stop attackers to infer

Manuscript received June 16, 2017; revised October 1, 2017. This work was supported in part by the National Nature Science Foundation of China (61762059), Youth Science and Technology Project of Gansu Province, China (1310RJYA004).

Yan Yan is with the School of Electrical and Information Engineering, Lanzhou University of Technology, Lanzhou, 730050, China (phone: +86-931-2976010; e-mail: yanyan@ lut.cn).

Wanjun Wang is with Lanzhou University of Arts and Science, Lanzhou, 730000, China (e-mail: wangwanjun1@163.com).

Xiaohong Hao and Lianxiu Zhang are with Lanzhou University of Technology, Lanzhou, 730050, China (e-mail: haoxh@163.com, zhanglianxiu_007@163.com).

sensitive information of target individual by linking some QIs together.

In order to avoid such linking attacks, the concept of K-anonymity [7] [8] was proposed, and many algorithms have been developed based on K-anonymity rules. These kinds of methods use generalization techniques to transform the values of QIs into less specific forms. Therefore, tuples can be divided into some QI-groups and none of them can be distinguished by QI-values within the same QI-group. From this way, privacy has been protected in a certain extent. How to select proper QIs is the primary problem for privacy preserving data publishing method based on K-anonymity model. In principle, QIs should include all the attributes that attackers may obtain from available databases and can be used for "linking attack". But for data publisher, it is unable to grasp all the detailed characteristics of external data, and cannot speculate all the background knowledge possessed by attackers. Most of the existing methods assume that the data publisher knows which are the QIs, or choose QIs according to some personal experiences. Some of the methods even select all the attributes except identifiers and sensitive attributes. Such above methods have some defects in different degrees. If the set of QIs contains too many attributes, the loss of information caused by generalization will be exacerbated. In some extreme cases, all the tuples in the table are generalized into one QI-group, which will seriously affect the availability of anonymous data. If the QIs are selected just based on some personal experiences, accuracy is difficult to be ensured and "linking attack" may occur and lead to the failure of anonymity protection. In addition, for different publishing data, the set of attributes used to be associated with external information may be different; even for the same data, external information can be different according to different publishing time and environment. Therefore, identification method for QIs should be a dynamic and changing process which is different according to the publishing data and external information. In this paper, we consider the problem of finding proper QIs for privacy preserving data publishing method based on K-anonymity model.

The rest of this paper is organized as follows. Section II reviews some previous research work related to the selection of QIs. Section III introduces the main idea to determine QIs by finding the set of cut-vertexes. The underlying concept of attribute graph and its generation method are proposed in this part. Section IV discusses the partitioning method of QIs for undirected and directed attribute graph respectively, and extends the method to solve the case of data publishing with multiple sensitive attributes. Section V compares and analyzes the size of QIs and computational complexity of the proposed algorithm with some existing methods. Section VI is the conclusion of the paper.

II. Related work

The research of privacy protection for centralized data publishing mainly focus on privacy rules [7-10], anonymity algorithms [11-13], improving the utility of anonymous data [14-16], etc. Some new studies have greatly expanded and improved applications of privacy preserving data publishing technology. Such as anonymity methods for multiple sensitive attributes [17][18], republish and dynamic update of anonymous data [19][20], and protection methods for the privacy of transactional information, social network, location and track information [21-23]. However, most of these studies ignored the problem of identification of QIs, or simply assume that they can separate sensitive attributes from QIs. For example, Shi et al. [24] introduced a new type of attribute "quasi-sensitive attributes" which are not sensitive by themselves, but may become sensitive when used in combination. Sei et al. [25] refine the classification of attributes into "explicit identifiers", "non-sensitive QIs", "sensitive QIs", "non-QI sensitive attributes", and "non-QI non-sensitive attributes". Although they have noticed that treating attributes that have a feature of both QIs and sensitive attributes are important, no methods have been put forward to clearly distinguish these attributes.

Motwani and Xu [26] use the concept of "separation ratio" and "distinct ratio" to quantitatively describe the distinguish ability of attributes. The "distinct ratio" measures distinguish ability of certain attribute by the ratio of the number of tuples of different values and the total number of tuples. The "separation ratio" describes the distinguish ability by the ratio of numbers of tuple pairs that can be distinguished by certain attribute and the number of all possible tuple pairs. Because it is provably hard to find QIs of the minimum size, they relaxed the problem and developed efficient algorithms to find small QIs with provable size and separation/distinct ratio guarantees. Experimental results show that the proposed algorithms only require one pass over the table and have better space and time complexities than the traditional greedy algorithm. However, for a single data set, QIs consist of attributes that can uniquely identify a tuple, but for an open data publishing environment with huge amount of big data, the situation will be quite different. When the publishing data can be associated with external knowledge (published data or background knowledge that attackers may get from somewhere), the probability of linking attack will be greatly enhanced, and some attributes which are not (or not fully) belong to QIs in a single data set are likely to become QIs which will disclosure privacy information after combining with external knowledge. The method of finding QIs proposed by Motwani and Xu [26] only considered the publishing data, but ignored potential data connection risks. Therefore, it cannot guarantee the accuracy of QIs and is not able to prevent linking attacks.

LEE et al [27] analyzed the factors and probability of re-identification for medical records according to inferable QIs. They treated QIs as variables that allow re-identification via a connection to certain individual and point out that it is possible to infer QIs from some background knowledge. In order to avoid the invasion of patient's privacy, five factors have been selected to be QIs which affected the probability of re-identification and could be inferred from background knowledge. Simulation experiments adopted by this paper use the probability of re-identification to determine the most effective factors among the combinations of QIs. The first limitation of this paper is that the number of inferable QIs for re-identification of medical records may be more than five. The actual number of QIs may be different according to different time, situation, and types of publishing data. For another one, the paper only analyzed characteristics of attributes from medical record related with patients' privacy and it lacked a general method suitable for common publishing of data.

Kumar et al [28] put forward an assessment method for the classification of patients' QIs. Ensembles of several multi-label learning algorithms have been applied to predict the accuracy of classification of QIs (race and gender of patients). Three stages of experimental method have been carried out on the UCI diabetics dataset by using different multi-label classifiers and evaluation measures. Such as binary relevance (BR), classifier chains (CC), Bayesian classifier chains (BCC), etc. Although the experimental results showed that the best classifier achieves a high overall accuracy, the transformation method which used to transform the problem of multi-label classification into single label classifications in this paper has a high degree of complexity. What's more, it can be used to predict the accuracy of prediction of patient QIs, but how to get the set of QIs was not mentioned in this paper.

Song et al [29] discussed the issue of selecting QIs for the publishing of views and analyzed the composing characteristics of QIs with/without functional dependencies. They suggested that if there have no functional dependencies in the publishing views, the set of QIs should be composed by the public attributes among the views; while if there have some functional dependencies, the set of QIs should not only contain public attributes but also include the left attribute of functional dependency. This method required to estimate relevant views which were possible be associated with the publishing view, as well as the functional dependencies among various attributes. For data publishers, it was difficult to identify relevant views and the set of QIs in some practical applications, because they had little information except the publishing data and some part of published views.

[30] and [31] proposed new algorithms for finding the set of relevant views and QIs based on hypergraph. The method mapped the publishing view and the set of published views into a hypergraph, and converted the problem of finding the set of relevant views into the searching for all the paths between two given nodes. Identification method for QIs still uses the ideas proposed in [29]. Actual process of this method need to degenerate the hypergraph into a common graph and has a high computational complexity. Besides, the set of QIs get from this method may include too many attributes, which is not good for K-anonymity model based on generalizations (detiled analysis is given in the section of experiments and analysis).

In this paper, we study the problem of finding QIs for privacy preserving data publishing method based on K-anonymity model. We take into account the connectivity of QIs between publishing data and others, and put forward a new way to determine QIs from the independence of sets. Identification algorithm has been developed to get QIs by finding cut-vertex from attribute graph, which represents all the possible relevance of publishing data, published data and external knowledge. Compared with some existing methods, the proposed identification method for QIs has better effect on partition ability and lower computational complexity.

III. Identification of QIs based on Cut-vertex

For data publisher, it is relatively easy to determine sensitive attributes combines with the contents of publishing data. For example, salary in income statistics, specific disease in medical statistics, location or trajectory in traffic statistics, etc. However, it is very difficult to determine QIs only according to some intuitive experiences. Firstly, data publishers are not able to make detailed predictions of which published information and background knowledge may be associated with the publishing data. Secondly, attackers may not only get some QIs of the target individual according to external database, but also obtain some background information by other means so as to infer the sensitive information of target individual. Some existing privacy preserving methods [26-31] mainly focus on finding out as many as possible QIs which may related with published data and external information. But with the increase of background knowledge and diversify of attacks, these kinds of methods will lead to increasing size of QIs, as well as the aggravation of generalization and the decline of data utility.

We noticed that in the field of Markov network, there are some conclusions regard to independence [32] [33] (shown in Fig.1). If the set of node X_B "split" the set X_A and X_C into two different sets, then given the set of node X_B, the set X_A is independent from the set X_C. This property can be also described as: if any path which begins from one node of set X_A and ends in one node of set X_C contains at least one node of set X_B, it can be determined that when the set X_B is given, the set X_A and X_C are independent from each other.

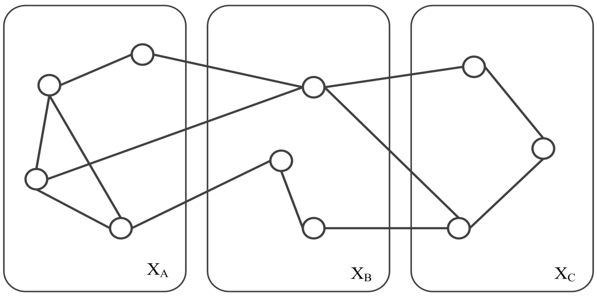


Fig.1 Example for the independence of Markov network

Based on the property of independence mentioned above and combine with the main idea of K-anonymity privacy protection method, we put forward the following assumption. If all the attributes can be expressed as nodes and all the relationships between attributes can be abstracted by edges, the publishing data together with the published data and external information can be represented by a graph. In this graph, X_C denotes the set of sensitive attributes that can be clearly determined. Then we can find out a set of node X_B from the rest of attributes so that the graph can be splited into three independent part (the third part is the set X_A). Therefore, take X_A and X_B to be the set of identifier attributes and QIs, any path from the node of X_A to the node of X_C should contain at least one node of X_B. For privacy preserving data publishing, it means when treating the set of node X_B as QIs and carrying out K-anonymity method, it can effectively prevent linking attacks and achieve the expectation of K-anonymity, no matter how much external knowledge can be associated with the nodes of X_A and X_B.

The identification method for QIs proposed in this paper firstly constructs attribute graph according to relationships between attributes. Then, the set of QIs will be determined by finding out cut-vertex on the paths from identifier attribute to sensitive attribute.

A. Attribute Graph

Definition 1 (Attribute Graph). Suppose all the attributes of data can be represented by nodes (denotes as $V=\{U_i\},i=1,2...m$) , and all the relationships between attributes can be represented by edges (denotes as $E=\{\varphi(e)\}=U_iU_j$, $i,j=1,2...m$), The resulting pattern is called attribute graph, denotes as $G_T=(V,E)$.

If there is no functional dependency between attributes, a pair of unordered nodes (U_i,U_j) $(i\neq j)$ can be used to indicate the relationship between this two attributes. Attribute graph formed by this way is called an undirected attribute graph. If there is some functional dependency between attributes, a pair of ordered nodes $<U_i,U_j>$ $(i\neq j)$ can be used to represent the dependency start from attribute U_i to attribute U_j , so as to form a directed attribute graph.

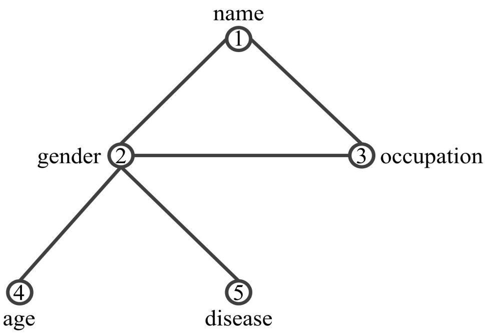


Fig.2 Undirected attribute graph

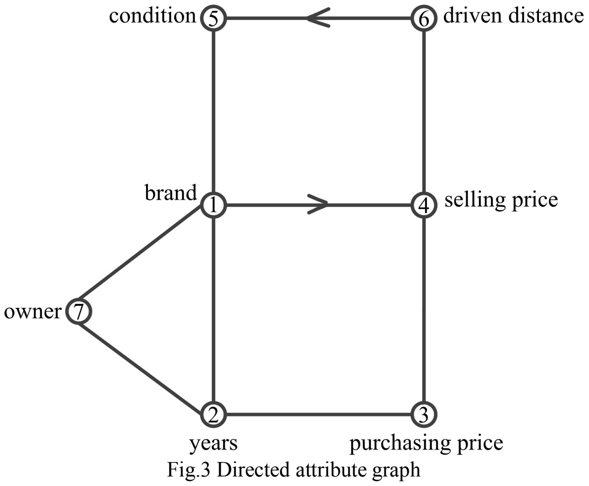


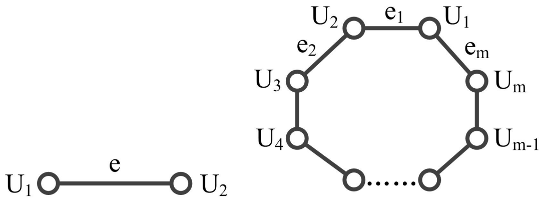
Fig.3 Directed attribute graph

Fig.2 is an example of undirected attribute graph constructed according to staff information and medical information. Suppose the released views are *V₁={gender, age}* and *V₂={gender, disease}*, the publishing view can be represented as *V₃={name, gender, occupation}*. Since there are no functional dependencies between *disease*, *gender*, *age* and *occupation*, all of the attributes form an undirected attribute graph. Fig.3 is the attribute graph for the sales of second-hand car and its background knowledge. Suppose the released information is *T₁={brand, years, purchasing price, selling price}* , and the publishing data can be expressed as *T₂={brand, condition, driven distance, selling price}*. Attackers may know that "someone has a car of the brand X, and it has been used for N years", so the background knowledge can be expressed as *T₃={owner, brand, years}*. Since the *brand* of a car and its *purchasing price* may affect the *selling price* to a certain extent, and the *condition* of a car will be affected by its *driven distance*. Therefore, the above information formed a directed attribute graph.

B. Generation Method of Attribute Graph

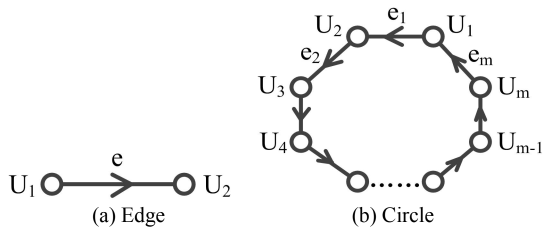
How to generate and represent attribute graph accurately and uniquely with the set of attributes and their relationships is the key point to affect the accuracy of identification of QIs. In this section, we put forward the generation method of attribute graph.

Attributes that appeared within one table have inherent characteristic of interconnection. $U_i(i=1,2...m)$ represents all the attributes that included in one table. For undirected attribute graph, if $i=2$, we can use U_1 and U_2 as well as the undirected edge $e=(U_1,U_2)$ between them to represent their interrelated characteristics, shown as Fig.4 (a). If $3\le i\le m$, a circle of non-redundant nodes and edges $U_1e_1U_2e_2...U_me_m$ (in which $e_1=(U_1,U_2)$, $e_2=(U_2,U_3)\cdots e_m=(U_m,U_1)$) can be used to connect all the nodes together, shown as Fig.4 (b). Any node within the circle is connected with its two neighbors and the relationships between each other can be transferred through the circle, so that any two attributes within one table can be connected with each other. For directed attribute graph, if $i=2$, we can use U_1 and U_2 as well as the directed edge $e=<U_1,U_2>$ to represent their interrelated characteristics, shown as Fig.5 (a). If $3\le i\le m$, we may firstly adjust the nodes with dependencies adjacent to each other, and then use a circle of non-redundant nodes and edges $U_1e_1U_2e_2...U_me_m$ (in which $e_1=<U_1,U_2>$, $e_2=<U_2,U_3>\cdots e_m=<U_m,U_1>$) to connect all the nodes together, shown as Fig.5 (b). Suppose the edges or circles appeared within one table can be denoted as subgraphs SG_i $(i=1,2...l)$. When the subgraph SG_u contains all the nodes and corresponding relationships of subgraph SG_v $(i\neq j)$, we say that SG_v is included by SG_u .



(a) Edge (b) Circle

Fig.4 Edge and circle for undirected attribute graph



(a) Edge (b) Circle

Fig.5 Edge and circle for directed attribute graph

Definition 2 (Relationship Matrix). Suppose the set of nodes for attribute graph G_T can be expressed as

$V=\{U_1,U_2...U_m\}$, relationship matrix can be defined as a matrix composed by element $a_{ij}(i,j=1,2...m)$.

$$a_{ij}=\begin{cases}1 & U_i \text{ and } U_j \text{ are in the same table}\\ 0 & U_i \text{ and } U_j \text{ are not in the same table or } i=j\end{cases} \tag{1}$$

For directed attribute graphs, some edges are composed by ordered pairs of nodes $<U_i,U_j>$, so that only one-way connectivity $U_i \to U_j$ can be formed. Therefore, compared with undirected attribute graphs, connectivity of nodes for directed attribute graphs is declined. Reflected in the relationship matrix, there are less locations with the value $a_{ij}=1$. For example, the undirected attribute graph and directed attribute graph shown in Fig.2 and Fig.3 may be expressed by relationship matrix R_{Ga} and R_{Gb} as follows.

$$R_{Ga}=\begin{bmatrix}0&1&1&0&0\\1&0&1&1&1\\1&1&0&0&0\\0&1&0&0&0\\0&1&0&0&0\end{bmatrix} \quad R_{Gb}=\begin{bmatrix}0&1&1&1&1&1&1\\1&0&1&1&0&0&1\\1&1&0&1&0&0&0\\1&1&1&0&1&1&0\\1&0&0&1&0&1&0\\1&0&0&1&1&0&0\\1&1&0&0&0&0&0\end{bmatrix}$$

Algorithm 1. **Attribute Graph**

***Initialization*：** publishing data, published data and external knowledge have been represent by the form of subgraphs.

***Termination condition*：** all the subgraphs have been processed or all the nodes have been included in the attribute graph.

***Input*：** relationship matrices $R_{SG_i}\ (i=1,2...l)$ of subgraphs.

***Output*：** relationship matrix R_G of attribute graph.

```
represent all tables by $R_{SGi}$
n = number of subgraphs
m = total number of attributes
Flag = zeros(n,1)
T ← select the subgraph with largest number of nodes
Flag(T) =1
$R_G$ ←$R_{SGT}$
NG=1
Node ← all the nodes within T
for i = 1: number of unprocessed subgraphs
    if $R_{SGi}$ contains Node completely
        Flag(i) =1
        NG= NG+1
    end
end
While (NG≠n) or (Flag≠1)
  P ← select unprocessed subgraph with largest number of nodes
  if size(P)>1
       P ← select the one with most repeated nodes with $R_G$
  end
  $R_G$ ←insert $R_{SGP}$ to $R_G$
  Flag(P) =1
  NG= NG+1
  Node ← all the nodes within $R_G$
     for  i = 1: number of unprocessed subgraphs
          if  $R_{SGi}$ contains Node completely
               Flag(i) =1
               NG= NG+1
          end
     end
end
return($R_G$)
```

Corollary: Under the premise of no changing of the number of attributes and their publishing relationships, different numbering order of attributes will not influence the structure of attribute graph. Therefore, relationship matrix R_G can be used to represent an attribute graph uniquely.

Proof: The “edge” and “circle” in the generation method of attribute graph have the property of symmetry. That means attribute nodes within the same edge/circle have the same importance, and connection relationships are equal for all the joint nodes. Therefore, the number of subgraphs and their connection relationships stay the same if there is no changing of the number of attributes and their publishing relationships. If the number of two nodes are swapped (denote as $U_i \leftrightarrow U_j$), the row *i* and *j* as well as the column *i* and *j* in the relationship matrix are all changed. Use R_G and R_G^* to represent the original relationship matrix and the corresponding relationship matrix after change. Elements of the row *i* and *j* are swapped, meanwhile, the elements of the column *i* and *j* are interchanged.

$$R_G=\begin{bmatrix}a_{11}&\cdots&a_{1i}&\cdots&a_{1j}&\cdots&a_{1m}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{i1}&\cdots&a_{ii}&\cdots&a_{ij}&\cdots&a_{im}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{j1}&\cdots&a_{ji}&\cdots&a_{jj}&\cdots&a_{jm}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{m1}&\cdots&a_{mi}&\cdots&a_{mj}&\cdots&a_{mm}\end{bmatrix} \quad R_G^*=\begin{bmatrix}a_{11}&\cdots&a_{1j}&\cdots&a_{1i}&\cdots&a_{1m}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{j1}&\cdots&a_{jj}&\cdots&a_{ji}&\cdots&a_{jm}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{i1}&\cdots&a_{ij}&\cdots&a_{ii}&\cdots&a_{im}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{m1}&\cdots&a_{mj}&\cdots&a_{mi}&\cdots&a_{mm}\end{bmatrix}$$

According to the definition of elementary transformation for matrix, if a matrix *A* can be transformed into a matrix *B* by a finite number of elementary row transformations, then the matrix *A* is equivalent to the matrix *B* (denoted as $A \overset{r}{\sim} B$). If a matrix *A* can be transformed into a matrix *B* by a finite number of elementary column transformations, then the matrix *A* is also equivalent to the matrix *B* (denoted as $A \overset{c}{\sim} B$). Therefore, if $U_i \leftrightarrow U_j$, $R_G \sim R_G^*$.

$$R_G \overset{r_{ij}}{\sim} \begin{bmatrix}a_{11}&\cdots&a_{1i}&\cdots&a_{1j}&\cdots&a_{1m}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{j1}&\cdots&a_{ji}&\cdots&a_{jj}&\cdots&a_{jm}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{i1}&\cdots&a_{ii}&\cdots&a_{ij}&\cdots&a_{im}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{m1}&\cdots&a_{mi}&\cdots&a_{mj}&\cdots&a_{mm}\end{bmatrix} \overset{c_{ij}}{\sim} \begin{bmatrix}a_{11}&\cdots&a_{1j}&\cdots&a_{1i}&\cdots&a_{1m}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{j1}&\cdots&a_{jj}&\cdots&a_{ji}&\cdots&a_{jm}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{i1}&\cdots&a_{ij}&\cdots&a_{ii}&\cdots&a_{im}\\ \cdots&\cdots&\cdots&\cdots&\cdots&\cdots&\cdots\\ a_{m1}&\cdots&a_{mj}&\cdots&a_{mi}&\cdots&a_{mm}\end{bmatrix}=R_G^*$$

The above situation can be extended into the case where multiple numbers of node are changed. According to the transitivity of elementary transformation, if $A \sim B$ and $B \sim C$, then $A \sim C$, it is easy to get $R_G \sim R_G^*$. □

Suppose the attribute graph G_T is composed by *n* subgraphs and contains a total number of *m* nodes (attributes). Running time of *Algorithm 1* is mainly spent on selecting a subgraph and determining whether the connection relationships of the unprocessed subgraphs have already been included. In the worst situation, all the subgraphs have to be selected and judged one by one. Once a subgraph has been selected, it has to be compared with the rest unprocessed ones. Therefore, the worst time complexity of *Algorithm 1* is $O(nm)$.

IV. Definition and Identification of QIs

Definition 3 (Set of QIs). For a given attribute graph G_T, the set of explicit identifiers can be expressed as U^{EI}, and sensitive attributes $U_i^{SA}\ (i=1,2...d)$ can be represented by the set V_{SA}. If the rest attribute nodes can be divided into two non-empty set V_N and V_{QI}, and the following conditions are satisfied, then V_{QI} is called the set of QIs for attribute graph G_T.

a) $V_N \cap V_{QI} = \phi$, $V_{QI} \cap V_{SA} = \phi$, $V_N \cap V_{SA} = \phi$,

$$V_N \cup V_{QI} \cup V_{SA} = V(G_T); \tag{2}$$

b) $U^{EI} \in V_N$, $U_i^{SA} \in V_{QI}$, for $\forall e = (U^{EI}, U_i^{SA})$ there is

$$e_i \cap e_j \subset V_{QI}. \tag{3}$$

According to Definition 3, the set of QIs V_{QI} is the necessary and minimum “bridge” which connects explicit identifier with sensitive attribute. Any path starts from explicit identifier U^{EI} to sensitive attribute U_i^{SA} must have at least one node within the set V_{QI}. Therefore, K-anonymity operation carried out on the set V_{QI} can effectively avoid the leakage of privacy information caused by linking attacks. Therefore, the problem of finding QIs for attribute graph can be transformed into the problem of determining the set of cut-vertex. It should be note that, the concept “cut-vertex” mentioned here is the cutting point of connectivity for all the paths from U^{EI} to U^{SA}, rather than the “cut point” of structure in the theory of graph.

A. Set of QIs for attribute graph with single sensitive attribute

For the case of data publishing with single sensitive attribute, use U^{EI} and U^{SA} to denote explicit identifier and sensitive attribute. The set of QIs for attribute graph G_T can be got through the following steps.

Step 1: generate attribute graph (represented by relationship matrix R_G) according to the publishing data, published data and external knowledge by using *Algorithm 1*;

Step 2: find out all the possible paths (denoted as $path_i\ (i=1,2...d)$) start from U^{EI} to U^{SA} by using a depth-first searching algorithm;

Step 3: determine the set of cut-vertexes (denoted as V_{QI}) on every path by using *Algorithm 2*.

Algorithm 2. Cut-vertex
Input: relationship matrix R_G, explicit identifier U^{EI} (denoted by number a), sensitive attribute U^{SA} (denoted by number b).
Output: the set of cut-vertex V_{QI}.

```
$V_N$=a;  $V_{SA}$=b;  $V_{QI}$=$\Phi$
m= size($R_G$)
$N_V$=2
P $\leftarrow$ find all possible paths from a to b
n=size(P)
Flag=zeros(n,1)
While ($N_V \neq$m) or (Flag$\neq$1)
    L $\leftarrow$ select unprocessed path with minimum number of nodes
    if size(L)=1
        Node $\leftarrow$ other nodes within path L except for a and b
        if size(Node)=1
            $V_{QI}$ $\leftarrow$ Node
            flag(L)=1
            RL $\leftarrow$ select unprocessed path which contains Node
            flag(RL)=1
            $N_V$ = $N_V$ +size(Node)
        elseif size(Node)>1
            for i=1:size(Node)
                $V_{QI}$ $\leftarrow$ Node(i)
                if $\forall$ Li $\cap$ Lj = $V_{QI}$
                    $V_{QI}$ $\leftarrow$ Node(i)
                    flag(L)=1
                    RL $\leftarrow$ select unprocessed path contains Node(i)
                    flag(RL)=1
                    $N_V$ = $N_V$ +size(Node)
                end
            end
        end
    elseif size(L)>1
        Node $\leftarrow$ find the intersections of paths in L
        if size(Node)=0
            $V_{QI}$ $\leftarrow$ insert other nodes within L except for a and b
            flag(L)=1
            $N_V$ = $N_V$ +size(other nodes)
        elseif size(Node)=1
            $V_{QI}$ $\leftarrow$ Node
            flag(L)=1
            RL $\leftarrow$ select unprocessed path which contains Node
            flag(RL)=1
            $N_V$ = $N_V$ +size(Node)
        elseif size(Node)>1
            for i=1:size(Node)
                $V_{QI}$ $\leftarrow$ Node(i)
                if $\forall$ Li $\cap$ Lj = $V_{QI}$
                    $V_{QI}$ $\leftarrow$ Node(i)
                    flag(L)=1
                    RL $\leftarrow$ select unprocessed path contains Node(i)
                    flag(RL)=1
                    $N_V$ = $N_V$ +size(Node)
                end
            end
        end
    end
end
return($V_{QI}$)
```

The time cost of *Algorithm 2* mainly includes two parts. Firstly, it has to find out all the possible paths from explicit identifier to sensitive attribute by using a depth-first searching algorithm. Suppose the attribute graph G_T contains a total number of *m* nodes, the frequency of recursive calling the depth-first searching is determined by the number of nodes connected to the starting point. In the worst case, the starting point connects to all the other nodes, so the recursive calling need to be performed *(m-1)* times at most. Return procedure of the recursive calling has to judge the nodes included in the path, computational complexity of this part is also related to the number of connected nodes. In the worst case, there are *(m-1)* nodes connected with the current node, so that the worst time complexity for the depth-first searching algorithm is $O(m^2)$. Secondly, the while loop is used to determine QIs which satisfy the properties of cut-vertex. In the most complex case, the longest path may contain all the nodes within the attribute graph. At this moment, there may be $1 + C_{m-3}^1 + C_{m-4}^1 + ... + 1$ paths at most, and the path contains the same nodes can only appear for once. In the worst case, it is required to determine cut-vertex for each path separately. So the worst time complexity for *Algorithm 2* is $O(m^2+m^2)\approx O(m^2)$.

Example 1. Let's consider the undirected attribute graph in Fig.2, in which the explicit identifier attribute U^{EI}*={name}* and sensitive attribute U^{SA}*={disease}*. To ensure that the privacy of individual cannot be get by linking attack, some attributes should be selected to carry out K-anonymity processing in the publishing view V_3*={name, gender, occupation}*. Relationship matrix of this undirected attribute graph can be expressed as R_{Ga}. According to the identification method for the set of QIs shown in *Algorithm 2*, all the paths from explicit identifier to sensitive attribute are shown in Table I (attributes are represented by their corresponding numbers). The obtained set of cut-vertex which satisfies the characteristics of QIs is V_{QI}*={gender}*.

TABLE I
PATHS FROM "NAME" TO "DISEASE"

NO.	Nodes on the path
1	1 2 5
2	1 2 3 5

TABLE II
PATHS FROM "OWNER" TO "SELLING PRICE"

NO.	Nodes on the path
1	7 1 4
2	7 1 2 4
3	7 1 2 3 4
4	7 1 3 4
5	7 1 3 2 4
6	7 1 5 4
7	7 1 5 6 4
8	7 1 6 4
9	7 1 6 5 4
10	7 2 4
11	7 2 1 4
12	7 2 1 3 4
13	7 2 1 5 4
14	7 2 1 5 6 4
15	7 2 1 6 4
16	7 2 1 6 5 4
17	7 2 3 4
18	7 2 3 1 4
19	7 2 3 1 5 4
20	7 2 3 1 5 6 4
21	7 2 3 1 6 4
22	7 2 3 1 6 5 4

Example 2. Considering the directed attribute graph for the sales of second-hand cars shown in Fig.3, in which the explicit identifier U^{EI}*={owner}* and sensitive attribute U^{SA}*={selling price}*. Data publisher do not want people to get the inference that "someone's car has sold for some price". Therefore, some QIs should be selected to carry out K-anonymity processing in the publishing table T_2*={brand, condition, driven distance, selling price}*. Relationship matrix of this directed attribute graph can be expressed as R_{Gb}. According to the identification method for the set of QIs shown in *Algorithm 2*, all the paths from explicit identifier to the sensitive attribute are shown in Table II (attributes are represented by their corresponding numbers). The obtained set of cut-vertex is V_{QI}*={brand, years}*. Combined with the published data T_1 and background knowledge T_3, it can be found that the quasi-identifier "years" has already been exposed. Therefore, even all the attributes in table T_2 are selected and protected by K-anonymity processing, attackers still can obtain the real *selling price* by linking attack. In order to realize data publishing without privacy disclosure, K-anonymity processing should be carried out on attribute *brand* and *years* in the published data T_1 and the publishing data T_2 at the same time.

B. Set of QIs for attribute graph with multiple sensitive attributes

The above discussion assumes that there is only one sensitive attribute in publishing data. However, during the process of actual data publishing may be more than one sensitive attribute need to be protected. Take medical information for example, it may include "diagnoses", "expenses" and other sensitive attributes which are highly associated with privacy. Therefore, it is necessary to discuss identification method of QIs for attribute graph with multiple sensitive attributes.

Actually, identification method of QIs for attribute graph with multiple sensitive attributes can be get by expanding the method for single sensitive attribute. For undirected attribute graph, if the publishing table contains multiple sensitive attributes $\{U_1^{SA},U_2^{SA}...U_n^{SA}\}$, all the sensitive attributes should appear on the circle formed by attributes connected one by one according to the construction method of attribute graph. Any two nodes located on the circle can connect with each other, therefore, the path which can reach sensitive attribute U_i^{SA} will also be able to reach sensitive attribute $U_j^{SA}\ (i\neq j)$. Based on this condition, multiple sensitive attributes $\{U_1^{SA},U_2^{SA}...U_n^{SA}\}$ can be merged into one node U_0^{SA}, so that all the relationships between sensitive attributes and other nodes will be converged on the node U_0^{SA}. Thus, the problem of finding QIs for attribute graph with multiple sensitive attributes can be transformed into the issue for single sensitive attribute (shown in Fig.6 and Fig.7).

For directed attribute graph with multiple sensitive attributes, different connection relationship can be formed due to different dependencies between attribute nodes, such as inclusion, subordination, derivation, etc. So it is unable to ensure the path which can reach sensitive attribute U_i^{SA} will also be able to reach sensitive attribute $U_j^{SA}\ (i\neq j)$. However, it may be possible to get the set of cut-vertex for each sensitive attribute $U_i^{SA}\ (i=1,2...n)$. So the problem of finding QIs for attribute graph with multiple sensitive attributes can be solved by using the method of single sensitive attribute for many times. Use $V_{QI}^i\ (i=1,2...n)$ to denote the set of QIs for single sensitive attribute U_i^{SA}, the final set of QIs can be expressed as:

$$V_{QI} = \bigcup_{i=1}^{n} V_{QI}^{i} \qquad (4)$$

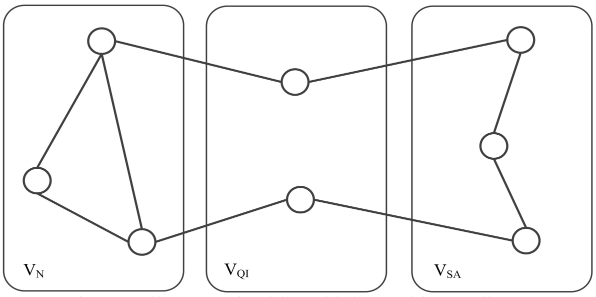


Fig.6 Attribute graph with multiple sensitive attributes

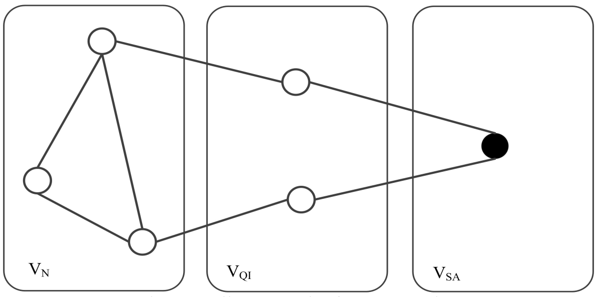


Fig.7 Attribute graph after conversion

V. Experiments and analysis

In this section, the analysis and experiments of the proposed identification method for QIs is given. Moreover, its partition ability and computational complexity is compared with some other related methods mentioned earlier.

A. Partition ability

We consider partition ability from two aspects. The first is whether a method takes into account the potential association relationships between publishing data and external knowledge. As mentioned in the former section, different publishing data can be associated with different external information by different attributes, which leads to different extent of privacy leakage. External information is changing according to different time and environment even for the same publishing data. When the publishing data is associated with some external knowledge, some attributes which are not (or not fully) belong to QIs are likely to become QIs in the open environment. So the probability of linking attack will be greatly enhanced. Therefore, it is important for the identification method to have the ability of evaluating potential association relationships between publishing data and external knowledge. For the second one, the size of QIs has a great impact on the performance of privacy preserving algorithm. Too many QIs will aggravate the loss of information caused by generalization in K-anonymity model, and drag the running time of algorithm at the same time. In some extreme cases, except explicit identifiers and sensitive attributes, all the rest attributes are selected to be QIs and all the tuples are generalized into one QI-group. This will lead to the failure of K-anonymity model directly. In this section, the separation/distinct ratio method [26], the re-identification method based on probability [27], and the method based on hypergraph [31] are selected and compared with the proposed identification method based on cut-vertex from the above mentioned two aspects.

[26] discuss the problem of finding minimum QIs by using the “distinct ratio” and “separation ratio”. Greedy algorithms for (ε,δ) separation/distinct minimum QIs have been put forward. For a publishing table with n tuples and m attributes, the algorithm chooses $k = \log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta}$ pairs of tuples randomly and each attribute maps to a subset of the k pairs separated by this attribute. After that, a greedy set cover algorithm is applied to find an exact set cover for those k pairs of tuples, and output corresponding attributes as QIs. The algorithm outputs a $(1-\varepsilon)$ separation QIs with the size of $(1+\ln\log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta})|I^*|$, where $|I^*|$ is the smallest key. For a publishing table, the size of key may be different from one attribute to m attributes. Therefore, the size of QIs identified by this method is about $1+ln\,log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta}$ to $m(1+ln\,log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta})$. However, the proposed method didn’t take into account the potential correlations between publishing data and external knowledge. So the accuracy of the selected QIs will be greatly reduced.

[27] point out that de-identified data can be re-identified from inference by using background knowledge. After analyzing some de-identification and re-identification techniques, five factors are selected as the QIs for medical records, which will affect the probability of re-identification and could be inferred from background knowledge. Although the method considers much information related with medical records, such as admit information, diagnosis, length of stay, provider license, etc., it is not enough for the open environment of data publishing with huge amount of information and dozens of attributes. Many practical incidents show that public information gathered from consumption, social network, education, social security, etc. can be linked together with medical records and therefore lead to the disclosure of people’s privacy. Besides, the number of QIs selected by this article is also limited. It will be variant from only one attribute to all of the five attributes. Actually, the number of inferable QIs for re-identification of medical records may be more than five. The coverage of the attributes may also beyond the five aspects mentioned in this paper.

The algorithm for finding QIs proposed in [31] mainly includes two steps. First of all, the set of relevant views have to be found out, which can connect the publishing view with some published views so as to infer secret information. Use $V'(X)$ and $V = \{V_1(U_1), V_2(U_2)...V_n(U_n)\}$ to denote the publishing view and the set of published views, $V'(X)$ and V can be mapped into a hypergraph. All the paths from explicit identifier U^{EI} to each of the node in publishing data X can be find out (denoted as RE), as well as all the paths from the sensitive attribute U^{SA} to each of the node in the publishing data X (denoted as RE'). So that the set of relevant views can be determined as $(RE \cup RE') - X$. Next, the set of QIs can be identified by selecting and merging the public attribute of the publishing view and each of the relevant view, so that the final result can be expressed as

$QI_{V'} = \bigcup_{i=1}^{l}(X \cap V_i)$. Suppose the publishing view contains a total number of m attributes, there will be at least one public attribute and at most $(m-1)$ intersection attributes between $V'(X)$ and each of the relevant view. In the worst case, all the attributes in publishing view may be selected into the set of QIs. The more the number of attributes included in publishing view, the more the number of QIs. This means more computational complexity for the algorithm that achieves K-anonymity protection by generalization operations.

Identification method for QIs proposed in this paper also includes two steps. Firstly, all the paths from explicit identifier to sensitive attribute have to be found out by a depth-first searching algorithm carried out on the relationship matrix R_G. Secondly, cut-vertex of all the paths need to be selected to get the final set of QIs. Generally, explicit identifier U^{EI} and sensitive attribute U^{SA} are not allowed to appear within the same table at the same time. According to the generation method of attribute graph proposed in this paper, explicit identifier U^{EI} and sensitive attribute U^{SA} will not appear in the same edge or circle. There will be at least one intermediate node and at most $(m-2)$ intermediate nodes on the path from U^{EI} to U^{SA} (shown in Fig.8). Let's consider the same situation as [31], suppose there are l tables (including the background knowledge) besides the publishing data X, and for each $X \cap V_i$ $(i = 1,2...l)$ there are at most $(m-1)$ intersections. According to the generation method of attribute graph proposed in this paper, we can obtain a circle with at most $(m-1)$ nodes, and there are l nodes have a separate edge. If the explicit identifier U^{EI} (or the sensitive attribute U^{SA}) locates on the end of edge, the sensitive attribute U^{SA} (or the explicit identifier U^{EI}) will locate on the circle or the end of other edge. There will be at least one interval node and at most $\left\lfloor \frac{m-1}{2} \right\rfloor$ interval nodes between them (shown in Fig.9). It is not difficult to find that compared with the method used in [31], the proposed identification algorithm based on cut-vertex has smaller size of QIs under the same circumstance.

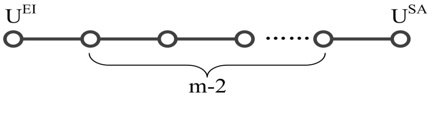


Fig.8 Attribute graph formed by linking of edges

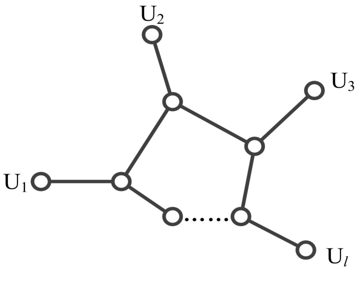


Fig.9 Attribute graph formed by circle and edges

Table III shows the partition ability of the methods discussed above. Among the methods consider potential connectivity of publishing data and external knowledge, identification method based on cut-vertex proposed in this paper has a smaller size of QIs. [27] uses the method to estimate the combination of QIs under a fixed number of prerequisites. It lacks the flexibility applicable for various data publishing environments, and has poor accuracy in the set of QIs. The method used in [31] only noticed the phenomenon of attributes linking, which is probably happened in various data tables or views, and takes as much associated attributes as possible into the set of QIs. However, the roles of associated attributes and their importance are not further distinguished. Therefore, the more the tables or views are associated with others, the more the attributes included in the table, the larger the final set of QIs. In this paper, we define QIs as the necessary and minimum "bridge" from explicit identifier U^{EI} to sensitive attribute U^{SA}. The proposed identification algorithm based on cut-vertex aims to find out the "must" rather than "all" associated attributes, by further differentiating the roles and importance of associated attributes. Hence the number of QIs has been reduced effectively. Fewer elements of QIs can significantly reduce the computational complexity of K-anonymity algorithm, and is also helpful to reduce the loss of information caused by generalizations, as well as improve the availability of anonymous data. Therefore, the proposed method has superior accuracy and partition effect than others.

TABLE III
COMPARISON OF PARTITION EFFECT

Methods	Connectivity	Minimum Size of QIs	Maximum Size of QIs
Ref.[26]	NO	$1 + ln\,log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta}$	$m(1 + ln\,log_{\frac{1}{1-\varepsilon}} \frac{2^m}{\delta})$
Ref.[27]	Yes	1	5
Ref.[31]	Yes	1	m
Ours	Yes	1	m-2

B. Computational Complexity

It has been proved that the problem of finding minimum key is an NP-hard problem, and the best-known approximation algorithm for this problem is a greedy algorithm, which has an approximation ratio of $O(\ln n)$ (where n represents the number of tuples). In view of this, [26] relaxes the problem of finding minimum QIs to the (ε, δ) separation/ distinct minimum QIs problem, which aims to find QIs with a small size such that, with probability at least $1-\delta$, the output QIs has separation/distinct ratio at least $1-\delta$. By sacrificing some accuracy on the result, [26] developed efficient algorithms that find small size of QIs with time complexities sublinear in the number of tuples. When taking ε and δ as constants, the approximation ratio of the proposed method will be $O(\ln m)$ (where m represents the number of attributes), which is smaller than $O(\ln n)$ for the traditional greedy algorithm when $n >> m$. The running time of the above algorithm is $O(m^4)$.

[27] uses a comparative analysis of the probability of re-identification according to the type and the range of inference. The method shown in formula (5) is used for measuring the probability of re-identification, where f refers to the size of QI-group, j is the number of QI-group in data set, θ represents the probability of re-identification. For example, if all the tuples can be separated into q QI-groups according to a quasi-identifier attribute (or a certain combination of QIs), the probability of re-identification will be $1/q$. When using this comparative method, we have to firstly get a set of all the combinations composed by inferable attributes. In [27], the set contains $C_5^0+C_5^1+C_5^2+C_5^3+C_5^4+C_5^5=32$ possible combinations. Then, the publishing data need to be scanned for many times in order to get QI-groups according to certain combination of attributes. For a publishing dataset with n tuples and m QIs, it needs at least an approximation ratio of $O(\ln n)$ for each scan. Therefore, the total running time of this algorithm will be $\sum_{i=0}^{m} C_m^i \times O(n) \approx 2^m O(n)\cdot$

$$\theta_j = \frac{1}{f_j} \tag{5}$$

The algorithm of finding QIs proposed in [31] firstly maps all the publishing views and published views into hypergraph. In order to get all the paths between a pair of points in the hypergraph, every two nodes within a hyper-edge are connected by a line where the number of nodes is equal to or more than three. So the hypergraph is degenerated into a common graph. Suppose the total number of nodes is m , the algorithm for the set of relevant views has a complexity of $O(m^5)$. Next, QIs are determined according to whether there have functional dependencies between attributes. Suppose the set of relevant views has a number of l views, if there are no functional dependencies among the attributes, the algorithm for finding the intersection part of views will be looped for l times. That is to say, the total complexity of the process for finding QIs is $O(l+m^5)\approx O(m^5)$ under the circumstances with no functional dependencies. When there are some functional dependencies between attributes, some judgments have to be carried out on the basis of previous algorithm, the overall complexity of the algorithm is also similar to $O(l+m^5)\approx O(m^5)$.

Identification algorithm for QIs based on cut-vertex proposed in this paper firstly represents all the publishing data, published data and other available external information by the form of undirected/directed attribute graph. Connections among attributes such as dependency, connectivity, subordination, etc are described by the relationship matrix. Suppose the attribute graph has a total number of m nodes, the identification algorithm for QIs mainly includes two steps: Firstly, all the possible paths from explicit identifier U^{EI} to the sensitive attribute U^{SA} are obtained by a depth-first searching strategy according to relationship matrix. Its complexity is about $O(m^2)$. Secondly, the set of QIs is determined by selecting cut-vertex from the above-mentioned paths, complexity of this part is also $O(m^2)$. So the identification algorithm for QIs based on cut vertex proposed in this paper has a time complexity of $O(m^2+m^2)\approx O(m^2)$ in the worst case.

TABLE IV
COMPARISON OF RUNNING TIME

Methods	Running Time
Ref.[26]	$O(m^4)$
Ref.[27]	$2^m O(n)$
Ref.[31]	$O(m^5)$
Ours	$O(m^2)$

Table IV compares the computational complexity of the methods mentioned above. The identification method for QIs based on cut-vertex proposed in this paper has lower computational complexity than others. The main reason lies in the following aspects: Firstly, the methods proposed in [26] and [27] use greedy or exhaustive algorithm to get all the possible QIs or combinations of QIs. It requires multiple scans of the publishing data and consumes a lot of computing time. [31] and our method resolve the problem of finding QIs by using the theory of graph. There is no need to scan the publishing data, therefore saving a lot of running time because the number of attributes is far less than the number of tuples. Secondly, compared with the method proposed by [31], we use circle instead of hyper-edge, which not only reserves the connectivity between attributes but also saves the process of transforming hypergraph into common graph. Thirdly, in order to determine QIs for the publishing view, [31] needs to search all the possible paths from explicit identifier and sensitive attribute to each of other nodes within the publishing view. In this paper, a depth-first searching strategy is used based on relationship matrix, which effectively reduced the complexity of path searching from $O(m^4)$ to $O(m^2)$. Finally, the partitioning method based on cut-vertex put forward in this paper selects attribute that satisfies the demand of cut-vertex and eliminates the path that includes the attribute at the same time, in order to avoid the traversal of all the possible paths and speed up the selection process.

VI. CONCLUSION

Automatic discovery and accurate classification of QIs are important factors for the success of privacy preserving data publishing methods. In this paper, we aim to discuss the role of QIs and the problem of finding QIs for K-anonymity model. We are inspired by the concept of independence from Markov network, and our contributions can be summarized as follows. Firstly, we put forward a new definition for QIs from the aspect of independence of sets. This new definition establishes the theoretical basis for the identification of QIs, that is, QIs should separate sensitive attributes from the other non-sensitive attributes. Secondly, we design a method to form directed/undirected attribute graph according to relationship matrix, which helps to reflect potential relationships among attributes. Finally, we put forward an efficient algorithm to identify QIs based on cut-vertex. Compared with some existing methods, the proposed identification algorithm has better ability to reflect potential association relationships of publishing data and external knowledge. What's more, it has smaller size of QIs and lower computational complexity under the same circumstance.

REFERENCES

[1] The white house. "Consumer data privacy: in a networked world", https://www.whitehouse.gov/sites/default/files/privacy-final.pdf

[2] Meng Xiaofeng, Zhang Xiaojian. "Big data privacy management", *Journal of Computer Research and Development*, vol.52, no.2, pp265-281, 2015.

[3] European Commission. "Proposal on general data protection regulation", http://ec.europa.eu/justice/data-protection/document/review2012/com_2012_11_en.pdf

[4] Carlos Costa, Maribel Yasmina Santos. "Big Data: State-of-the-art Concepts, Techniques, Technologies, Modeling Approaches and Research Challenges", *IAENG International Journal of Computer Science*, vol.44, no.3, pp285-301, 2017.

[5] Yawar Abbas Bangash, Qamar ud Din Abid, Alshreef Abed Ali A, et al. "Security Issues and Challenges in Wireless Sensor Networks: A Survey", *IAENG International Journal of Computer Science*, vol.44, no.2, pp135-149, 2017.

[6] Hirofumi Miyajima, Noritaka Shigei, Hiromi Miyajima, et al. "New Privacy Preserving Back Propagation Learning for Secure Multiparty Computation", *IAENG International Journal of Computer Science*, vol.43, no.3, pp270-276, 2016.

[7] Sweeney L. "K-Anonymity: A model for protecting privacy", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol.10, no.5, pp 557-570, 2002.

[8] Sweeney L. "Achieving k-anonymity privacy protection using generalization and suppression", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol.10, no.5, pp571-588, 2002.

[9] Priyadarsini RP, Sivakumari S, Amudha P. "Enhanced l-Diversity Algorithm for Privacy Preserving Data Mining", *Proc. 51st Annual Convention of the Computer-Society-of-India*, India, 2016, pp14-23.

[10] Yamaoka Y, Itoh K. "k-Presence-Secrecy: Practical Privacy Model as Extension of k-Anonymity", *IEICE Transactions on Information and Sysyems*, issue 4, pp730-740, 2017.

[11] Casas-Roma J, Herrera-Joancomarti J, Torra V. "A summary of k-degree anonymous methods for privacy-preserving on networks", *Advanced Research in Data Privacy*, issue 567, pp231-250, 2015.

[12] Amiri Fatemeh, Yazdani Nasser, Shakery Azadeh, et al. "Hierarchical anonymization algorithms against background knowledge attack in data releasing", *Knowledge-based Systems*, vol.101, pp71-89, 2016.

[13] Le Junqing, Liao Xiaofeng, Yang Bo. "Full autonomy: A novel individualized anonymity model for privacy preserving", *Computers & Security*, vol.66, pp204-217, 2017.

[14] Nayahi JJV, Kavitha V. "Privacy and utility preserving data clustering for data anonymization and distribution on Hadoop", *Future Generation Computer System-The International Journal of Escience*, vol.74, pp393-408, 2017.

[15] B Palanisamy, L Liu. "Privacy-preserving Data Publishing in the Cloud: A Multi-level Utility Controlled Approach", *Proceedings of the IEEE 8th International Conference on Cloud Computing*, USA, pp130-137, 2015.

[16] Sanchez David, Domingo-Ferrer Josep, Martinez Sergio, et al. "Utility-preserving differentially private data releases via individual ranking microaggregation", *Information Fusion,* vol.30, pp1-14, 2016.

[17] Qinghai Liu, Hong Shen, Yingpeng Sang. "Privacy-Preserving Data Publishing for Multiple Numerical Sensitive Attributes", *Tsinghua Science and Technology*, vol. 20, no.3, pp246-254, 2015.

[18] Zhang L, Xuan J, Si RQ, et al. "An Improved Algorithm of Individuation K-Anonymity for Multiple Sensitive Attributes", *Wireless Personal Communications*, vol.95, no.3, pp2003-2020, 2017.

[19] Li Jin, Liu Zheli, Chen Xiaofeng, et al. "L-EncDB: A lightweight framework for privacy-preserving data queries in cloud computing", *Knowledge-Based Systems*, vol.79, pp18–26, 2015.

[20] Wang J, Zhang YH, Wang YY, et al. "RPRep: A Robust and Privacy-Preserving Reputation Management Scheme for Pseudonym-Enabled VANETs", *International Journal of Distributed Sensor Networks*, vol.12, issue 3, pp1-15, 2016.

[21] Dargahi Tooska, Ambrosin Moreno, Conti Mauro, et al. "ABAKA: A novel attribute-based k-anonymous collaborative solution for LBSs", *Computer Communications*, vol. 85, pp1-13, 2016.

[22] Terrovitis Manolis, Poulis Giorgos, Mamoulis Nikos, et al. "Local Suppression and Splitting Techniques for Privacy Preserving Publication of Trajectories", *IEEE Transactions on Knowledge and Data Engineering*, vol. 29, no. 7, pp1466-1479, 2017.

[23] Dara E. Seidl, Piotr Jankowski, Ming-Hsiang Tsou. "Privacy and spatial pattern preservation in masked GPS trajectory data", *International Journal of Geographical Information Science*, vol.30, issue 4, pp785-800, 2016.

[24] P. Shi, L. Xiong, B. Fung. "Anonymizing data with quasi-sensitive attribute values", *Proceedings of the 19th ACM international conference on Information and knowledge management*, Canada, pp1389–1392, 2010.

[25] Yuichi Sei, Hiroshi Okumura, Takao Takenouchi, et al. "Anonymization of Sensitive Quasi-Identifiers for l-Diversity and t-Closeness", *IEEE Transactions on Dependable and Secure Computing*, DOI 10.1109/TDSC.2017.2698472

[26] Rajeev Motwani, Ying Xu. "Efficient Algorithms for Masking and Finding Quasi-Identifiers", *Proceedings of VLDB*, Vienna, Austria, pp758-769, 2007.

[27] Yong Ju LEE, Kyung Ho LEE. "Re-identification of medical records by optimum quasi-identifiers", *International Conference on Advanced Communication Technology (ICACT)*, Bongpyeong, South Korea, pp428-435, 2017.

[28] Naveen Kumar Parachur Cotha, Marina Sokolova. "Multi-label learning in classification of patients' quasi-identifiers", *Prog Artif Intell*, vol.4, pp37-48, 2015.

[29] Song Jinling, Huang Liming, Liu Guohua. "Algorithm for Finding Quasi-identifiers in the k-anonymity Method", *Journal of Chinese Computer Systems*, vol.29, no.9, pp1688-1693, 2008.

[30] Song Jinling, Liu Guohua, Huang Liming, et al. "Algorithm to Find the Set of Relevant Views and Quasi-Identifiers for k-anonymity Method", *Journal of Computer Research and Development*, vol.46, no.1, pp77-88, 2009.

[31] Huang L M, Song J L, Lu Q C, et al. "Hypergraph-based solution for selecting quasi-identifier", *International Journal of Digital Content Technology and its Applications*, vol.6, no.20, pp597-606, 2012.

[32] Santana R, Shakya S. "Probabilistic Graphical Models and Markov Networks", *Markov Networks in Evolutionary Computation*, vol.14, no.1, pp 3-19, 2012.

[33] Wang Feiyue, Han Suqing. "Probabilistic Graphical Models: theory and technology", *Tsinghua University Press*, 2015.