Incremental Anonymous Privacy-Protecting Data Mining Method Based on Feature Correlation Algorithm

Yongliang Jia[1](✉), Peng Tao[1], Dapeng Zhou[2], and Bing Li[1]

[1] State Grid Hebei Marketing Service Center, Shijiazhuang 050000, China
[2] Information and Communication Branch, State Grid Liaoning Electric Power Co., Ltd.,
Shengyang 110000, China

Abstract. With the arrival of the information age, privacy leaks have been getting increasingly serious. The commonly-used method for privacy protection is to introduce k-anonymity when data publishing. In this paper, how to employ k-anonymity in stream data sets is studied, and the relationship and the sensitivity matrix of the basic knowledge about Quasi identifier (QI) and sensitive attributes are established. RSLK-anonymity algorithm is proposed for solving the private information leakage during the streaming data publishing. The main idea is to anonymize the streaming data set on the basis of sliding window as well as relation and sensitive matrix, which can make anonymous streaming data effectively defense background knowledge attack and homogeneity attack while solving sensitive attribute diversity. Experimental results indicate that RSLK-anonymity algorithm is practical, effective and highly efficient.

Keywords: K-anonymity · Privacy protection · Sliding window · Algorithm of correlation feature

1 Introduction

In the field of conventional database, k-anonymity is a hot topic in protecting privacy, which was proposed by Samarati P and Sweeney L in 1998. In order to solve privacy leakage, k-anonymity model requires some unidentified individuals in the published data, which makes the attacker unable to differentiate the privacy of specific individuals and prevent individual privacy leakage. In recent years, k-anonymity has raised widespread concern in academia. Many scholars have studied privacy protection at different levels.

In many cases, transaction data presents the form of high-speed data stream [2, 3], such as wireless sensor networks. Therefore, it needs a solution that specifically considers the features of data flow. Data flow has the following features:

(1) The elements come online;
(2) The system cannot control the arrival sequence of data elements;
(3) Once the elements are seen or processed, it is not easy to retrieve or see them again, unless they are explicitly stored in memory.

Y. Tian et al. (Eds.): ICBDS 2020, CCIS 1415, pp. 204–214, 2021.
https://doi.org/10.1007/978-981-16-3150-4_18

This paper adopts the data structure of rule tree [3, 4] to complete the data k-anonymization in sliding window, and creates the correlation features and sensitivity matrix of background knowledge about QI and sensitive attributes. To solve privacy leakage in data stream publishing, an improved RSLK anonymity algorithm is proposed to protect the privacy of anonymous data stream.

2 Related Work

Many researchers try to combine privacy protection and data stream management to achieve data stream privacy protection.

Li Jianzhong etc. put forward a new approach called SKY (Stream K-anonymity) to promote the k-anonymity of data streams continuously. It considers the k-anonymity of data streams to protect privacy [2]. In [3], how to protect the privacy of users on the sliding window of transaction data stream is studied. It is challenging since the sliding windows are updating frequently and quickly. Later, a novel method called SWAF (Sliding Window Anonymization Framework) was proposed which can promote k-anonymity on sliding windows. SWAF has the following three advantages:

(1) The processing time of each tuple of data stream is very short;
(2) The memory requirement is very small;
(3) Privacy protection and the anonymous sliding window practicability are considered.

In [4], Zhang Junwei etc. studied KIDS (K-anonymity Data Stream Based on Sliding Window), and adopted continuous k-anonymity on sliding window. KIDS can protect the data stream privacy well, and consider the data distribution density in data stream, thus greatly improving the practicability of data.

Cao Jianneng etc. proposed CASTLE (Continuous Anonymization of Stream Data Through Adaptive Clusterering), which is a cluster based scheme, which can anonymize the data stream in real time, and ensure the novelty of anonymous data by meeting the certain delay constraint [5, 6].

Sylvia L. Osborn and Hessam zakerzadeh proposed a cluster based k-anonymity algorithm, which is named as FAANST (Fast Anonymous Algorithm for Numerical Stream Data), which can anonymize numerical stream data very quickly. As is shown by research, extending FAANST to support data streams composed of classification and numerical values is very easy [7].

Yang Gaoming etc. proposed a new k-anonymity data publishing method based on weak clustering. The practicability has two advantages: first, the processing time of each data stream tuple is very short. Second, less memory is needed [8].

However, few research has considered the relationship between sensitive attributes based on data flow and QI, sensitive attributes of their own features, diversity features of sensitive attributes, etc. [9–12].

3 Problem Definition

3.1 Anonymous Stream Data on Sliding Window

Here are some definitions of the terms mentioned in this paper.

Definition 1. Data stream (DS) is an infinite time series, and its increment order is DS{s_1, p_1,… s_n, p_n,…}; where s_i is a tuple with sequence number p_i, and $p_i < p_j$ indicates that p_i arrives before p_j. Each S_i contains vectors of *m* values (a_1, a_2,…, a_m), where a_i is derived from the finite field D_i [2].

Among many data stream mining research, to highlight the latest data stream and avoid storing potential infinite data stream in memory, researchers usually use sliding window method to process data. Sliding windows always keep the latest part of the data flow. As new tuples continue to enter from the data stream, the sliding window replaces the oldest tuple with a new one. The sliding window can be classified into two types, one is based on counting and the other is based on time. To make it simple, this paper describes our work with the sliding window based on counting.

Definition 2. Define <s_n, p_n> to be the latest tuple from data stream s, and the sliding window SW_l to be a subset of DS {<s_n, p_n>, … <s_{n-l}, p_{n-l}>}.

Definition 3. K anonymous data stream on sliding window. Suppose the sliding window SW_1 = {<s_n, p_n>,… <s_{n-l}, p_{n-l}>} and the QI attribute set Q, the data set ASW_1 = {<g_n, p_n>,…, <g_{n-l}, p_{n-l}>} is generated to make:

(1) ASW_1 meets the k-anonymity of Q;
(2) ∀I ∈[n−l, n], <g_i, p_i> is a tuple summary of <s_i,p_i>.

For the protocol of sliding window, when the sending window and receiving window sizes are fixed to 1, the protocol will degenerate into stop-and-wait protocol. According to that, the sender shall stop the follow-up action after sending a frame, and can continue to send the next frame only after the receiver has received the correct response. As the receiver shall determine whether the received frame is a newly sent or a resent frame, in each frame, the sender shall add a sequence number. Since the stop-and-wait protocol stipulates that a new frame can only be sent after one frame has been successfully sent, only one bit is enough to carry out numbering.

The following is the communication process when the 1-bit sliding window protocol is executed. In the triplet (i, j, k), i represents the number of the message sent by the sender (either A or B), j represents the number of the message received by the sender from the other party last time, and k represents the data. Figure 1 shows the execution of the sliding window protocol.

3.2 Correlation Feature and Sensitivity Matrix on the Basis of Background Knowledge

Domain experts, or direct analysis of basic data can provide background knowledge. which describes the impact of various sensitive attributes generated by QI attributes [9, 12].

The relationship and sensitivity matrix M|S is used to explain the impact level of sensitive attributes generated by QI attribute and sensitive attribute themselves:

- t_{ij}: The impact level of NO.j. OI sensitive attribute generated by NO.i.
- b_i: The weight of sensitivity attribute value of NO.i.

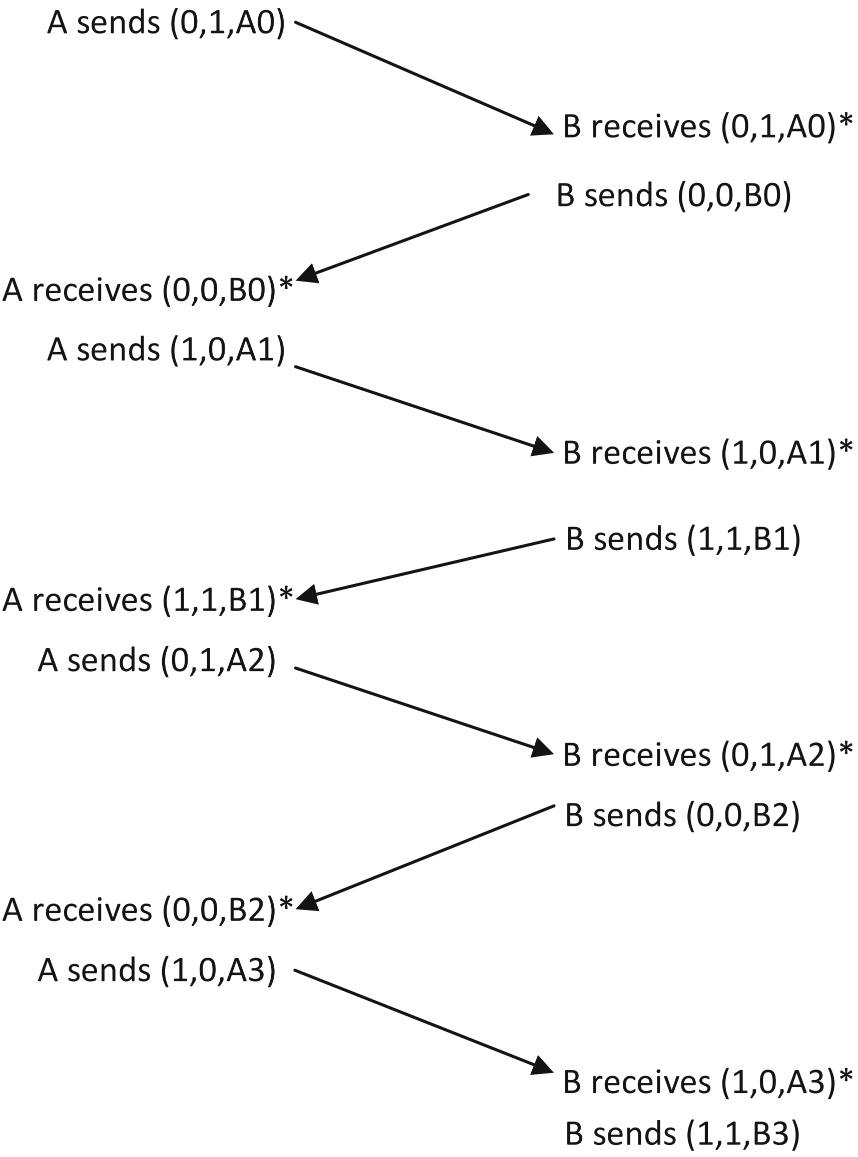

Fig. 1. Implementation process of sliding window protocol

There are n + 1 columns and m rows in the relationship and sensitivity matrix M|S, m is the quantity of sensitivity attribute, n is the QI attribute quantity, thus the sample matrix is:

$$M|S = (t_{ij}|b_i)_{m\times(n+1)} = \left[\begin{array}{cccc} QI_1 & QI_2 & QI_n & S \\ t_{11} & t_{12} & t_{1n} & |\, b_1 \\ t_{m1} & t_{m2} & t_{mn} & b_m \end{array}\right]$$

The weight values of t_{ij} and b_i are determined by experts or experience values. For instance, the weight of QI in the aforementioned formula can be divided into five grades:1, 0.8, 0.4, 0.1, 0, and the weight of S in the formula mentioned above is divided into five grades: 0.10, 0.30, 0.50, 0.80, 0.90. Taking diseases as an example, influenza is a common disease. Due to the features of local influenza, disease weight can be 0.11, ZIP weight can be 0.8, gender weight can be 0.2, etc. The obesity disease weight can be used as 0.12, and the weight of influenza and obesity can be used as 0.1, 0.01 and 0.02 to indicate disease. The respiratory disease weight is 0.31, the weight of AIDS, breast cancer and lung cancer are 0.93, 0.92 and 0.91. Different diseases shall not have the same disease weight values. Thus, the sensitivity matrix can be expressed as:

$$M|S = (t_{ij}|b_i)_{9\times(5+1)} = [\begin{array}{ccccc|c} \mathit{race} & \mathit{occupation} & \mathit{Birth} & \mathit{Sex} & \mathit{ZIP} & \mathit{Disease} \\ 0 & 0 & 0 & 0.2 & 0.8 & 0.11 \\ 0 & 0.1 & 0.4 & 1 & 0 & 0.92 \\ 0 & 0.1 & 0.4 & 1 & 0 & 0.92 \end{array}]$$

3.3 Overview of Feature Correlation Algorithm

3.3.1 Overview of Correlation Rule

Feature rule was proposed by Agrawal et al. in 1993, mainly to excavate the correlation between data. Suppose that the set of data items is A, the task set of database events is B, and each event I is the set of all A, then there is an identifier in the event of I $\subseteq$ A, which is indicated by IAB. Suppose M is a subset of A, if I contains M, if and only if M $\subseteq$ I. In this case, the correlation rule can be expressed as a form similar to M $\Longrightarrow$ N, where M $\subset$ A, N $\subset$ A, and M $\cap$ N = $\emptyset$.

The correlation rules of M $\Longrightarrow$ N can be quantified by the following two methods: support level (percentage of B contains both M and N at the same time, which is an evaluation of importance) and confidence level (percentage of B contains N while containing M, which is an evaluation of accuracy). Among them, the support level is s (M $\cup$ N), and the confidence level is C (N|M) = s (M $\cup$ N) * s (M).

A well-known story of correlation feature is "diapers and beer". This story shows that putting diapers and beer together will greatly increase the turnover of supermarkets. This is because men will buy beer for themselves when they buy diapers for their children after work, so putting diapers and beer together will increase the probability of purchasing. This is the reflection of the correlation feature. For example, taking diapers and beer as examples, the following table shows the categories of commodities, and the correlation rule is X $\Longrightarrow$ y (Table 1).

If the correlation rule is: {milk, diapers} $\Longrightarrow$ beer, then, the support level is:

$$s = \frac{\sigma(X \cup y)}{|T|}(s = P(X, y))$$

The confidence level is

$$c = \frac{\sigma(X \cup y)}{\sigma(X)}(c = P(y|X))$$

Table 1. Commodity list

No.	Category
1	Bread, milk
2	Bread, diapers, beer, eggs
3	Milk, diapers, beer, coke
4	Bread, milk, diapers, beer
5	Bread, milk, beer, coke

3.3.2 Feature Selection Algorithm Based on Correlation Rules

In recent years, feature selection algorithm has attracted widespread concern, especially in the field of pattern recognition and data mining. Many researchers have fully studied the feature algorithm, but the feature selection algorithm based on correlation rules is relatively rare. The research based on correlation rules can find the correlation between data items, and make full use of the association between attributes by selecting the best rules for the data.

The algorithm consists of three steps, including generating rule set, constructing attribute set and testing attribute set.

Step 1: Generate rule set: firstly, the training set is generated into a rule set, if necessary, the continuous attributes of non-class attributes are discretized, and then the rules selected from the rule set as class attributes are calculated to generate effective rule sets, and the support level and confidence level of the new rule set are calculated.

Step 2: Construct the attribute set. On the basis of the rule set generated in step 1, select the number of iterations to generate the attribute set.

Step 3: Test the attribute set. The generated attribute set needs to be evaluated to show the effectiveness of the feature selection algorithm based on correlation rules. The dataset is classified by the classifier, and the classification accuracy is given to evaluate the advantages and disadvantages of feature selection.

3.3.3 Rule Tree

Feature selection has always been a pattern recognition and an active research field in statistical and data mining aspects. The most important thought of feature selection is to select a subset of input variables through eliminating features with little or no prediction information. Feature selection can obviously enhance the comprehensibility of the classifier model, and usually establish a model that can be better extended to invisible points. In addition, under normal conditions, discovering the correct subset of prediction feature itself is a critical problem [13].

Rule tree is introduced in this paper. The domain general hierarchy is a tree. The root node of the tree is the value of Dn, and the leaf node is the set of values of D1. The edge from vi to vj ($i < j$) indicates that vj is a generalization of vi, in which $vi \in Di$, $vj \in Dj$.

3.4 Anonymization

Definition 4. RSLK anonymity. T (a_1,..., a_n) is a table if T satisfies K anonymous data stream on sliding window and the following conditions are satisfied:

(1) In FS (g), $\forall b_i < c$, all tuples in FS (g) are output directly. Otherwise, condition (2) and condition (3) must be met. Here, the threshold $c > 0$, b_i is the s column vector in the M|S, FS (g) is the buffer area of g, and $1 \le i \le |FS(g)|$,$|FS(g)| \ge k$.

(2) $L = \sum_{j=1,\ldots,|FS(g)|} count(|b_i, b_j| > 0)$, $1 \le i \le |FS(g)|$, the count b_i, b_j is the s column vector in the M|S, L is the number of different sensitive attribute values, which represents the sensitive attribute diversity.

(3) If in the count matrix M|S, $P = count(|MAX_{i=1,\ldots}|FS(g)|t_{ik}| = 1) > 0$, when t_{ik} is generated, the generalized hierarchy is promoted to the parent node or suppressed directly under the premise of anonymity.

4 RSLK Anonymous Algorithm

RSLK Anonymous Algorithm is described in details as follows.

Input: Flow *s*, rule tree S-Tree, relationship and sensitivity level matrix M|S based on background knowledge, threshold value *c*, parameter *k* of *k* anonymity.

Output: Anonymous tuples.

```
While (true)
    Read a new tuple $t$ from S;
    According to $P > 0$, search S-Tree from root to node $g$, which is the most specific node containing $t$;
    If ($g$. disease $< C$, G is the work node, $L > 0$)
        Anonymize tuple $t$ with $g$ and output it;
    else
        Insert $t$ into the buffer area FS($g$) of G;
        if ($L > 0$ and FS ($g$) satisfies k-anonymity)
            Anonymize and output FS($g$)with $g$;
            Mark $g$ as work node;
        end if
    end if
end while
```

In RSLK-anonymity algorithm, the nodes of rule tree are divided into 2 categories as candidate node and work node [2]. Candidate nodes do not meet the requirements of K anonymity. The work node satisfies the k-anonymity features. Each candidate node *g* is allocated with a buffer FS (g) as a holding area.

While the new tuple *t* reaches sliding window, RSLK-anonymous algorithm searches the specialized tree to determine the most specific generalized node *g* containing *t* based on P > 0. If *g* is a work node, the tuple *t* is anonymized with *g* based on g.disease < c and L > 0, which ensures the diversity of "diseases" of sensitive attributes and outputs them immediately. Otherwise, *g* is a candidate node, and *t* is stored in FS (g) until FS (g) meet the requirements of k-anonymity of definition 2, then output all tuples in FS (g), and candidate node *g* will become a working node.

5 Experiment Result

This experiment is performed on a PC (Windows 10, Intel Core i5 2.4 GHz CPU)and 4 GB main memory. The algorithm is implemented in Visual C++ 6.0. Each experiment was tested 10 times, and the average value of ten experiments was obtained.

In the experiment, adult census data sets from UC Irvine machine learning repository are used [14, 15]. The database has been used by many researchers and has become a benchmark in the field of data privacy. The race, occupation, age, gender and ZIP of adults are regarded as quasi identifier (QI), and the "disease" column is added as the sensitive attribute composed of {influenza, lung cancer, shortness of breath, obesity, breast cancer, AIDS}.

10000 tuples were randomly selected from adults as sliding window, and disease values were randomly assigned to each tuple based on basic medical knowledge.

5.1 Effective Time

Figure 2 shows the advantages of RSLK anonymous algorithm. The algorithm makes data publishing available and efficient by using c (the weight of sensitive attributes). From the experimental diagram, we can see that the larger the value of c, the more effective the running time is. This is because c represents the weight of sensitive attributes, and the data is anonymous through tuples, which can ensure the effective output of data and efficient and useful.

5.2 Running Time

As shown in Fig. 3, the running time increases with tuples numbers in the sliding window, since more time is needed by the algorithm to calculate and search tuples in the rule tree satisfying the anonymity condition. Because the computation level of tuples satisfying the anonymity condition increases, the running time increases with the increase of *k* value.

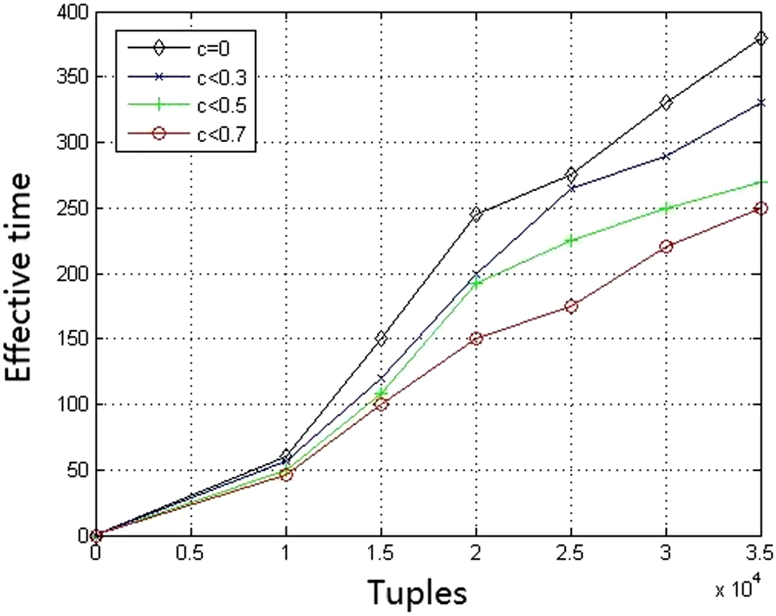

Fig. 2. Comparison of effective time when c changes

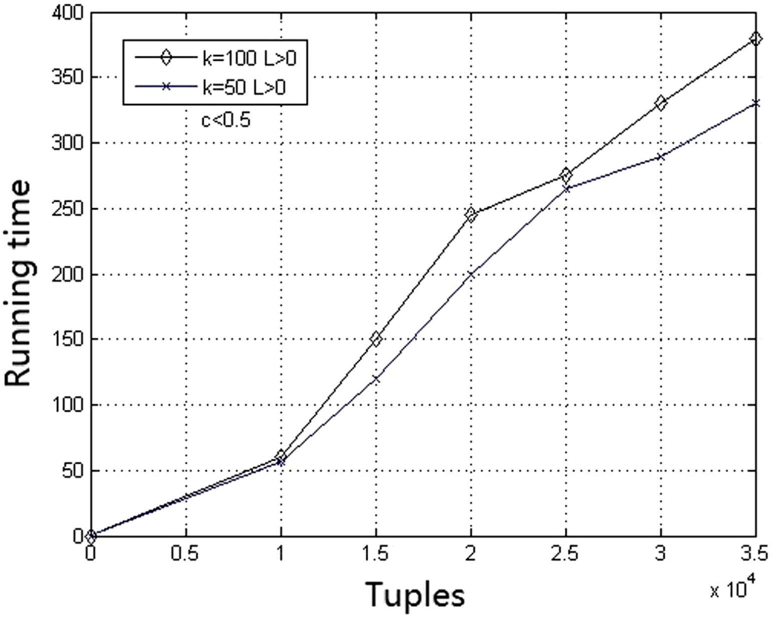

Fig. 3. Comparison of running time when the number of tuples changes

6 Conclusion

The main contribution of this paper is to put forward RSLK anonymity algorithm to avoid privacy leakage when stream data is published. The main thought is to anonymously process the background knowledge of stream data set in sliding window based on relation and sensitivity matrix. It can effectively defense the background knowledge attack and homogeneity attack in the process of stream data output, and solve the diversity of sensitive attributes. It's shown in the experimental results that RSLK algorithm is practical, effective and efficient, and its performance is better than that of k anonymous algorithm.

Acknowledgement. This paper is supported by the science and technology project of State Grid Corporation of China: "Research and Application of Key Technology of Data Sharing and Distribution Security for Data Center" (Grand No. 5700-202090192A-0-0-00).

References

1. Samarati, P., Sweeney, L.: Generalizing data to provide anonymity when disclosing information. In: Proceedings of the Seventeenth ACM SIGACT-SIGMOD-SIGART Symposium on Principles of Database Systems, p. 188 (1998)
2. Li, J., Ooi, B.C., Wang, W.: Anonymizing Streaming Data for Privacy Protection (2008)
3. Wang, W., Li, J.: Privacy protection on sliding window of data streams, Collaborative Computing: Networking, Applications and Worksharing, 12–15 November 2012, pp. 213–221 (2012)
4. Zhang, J., Jing, Y., Zhang, J., et al.: An improved RSLK-anonymity algorithm for privacy protection of data stream. Int. J. Adv. Comput. Technol. **4**(9), 218–225 (2012)
5. Cao, J., Carminati, B., Ferrari, E., et al.: CASTLE: continuously anonymizing data streams. IEEE Trans. Dependable Secure Comput. **8**(3), 337–352 (2011)
6. Cao, J., Carminati, B., Ferrari, E., et al.: CASTLE: A delay-constrained scheme for ks-anonymizing data streams. In: IEEE International Conference on Data Engineering. IEEE (2008)
7. Zakerzadeh, H., Osborn, S.L.: FAANST: fast anonymizing algorithm for numerical streaming DaTa. In: Garcia-Alfaro, J., Navarro-Arribas, G., Cavalli, A., Leneutre, J. (eds.) Data Privacy Management and Autonomous Spontaneous Security. Springer, Heidelberg (2011). https://doi.org/10.1007/978-3-642-19348-4_4
8. Yang, G., Yang, J., Zhang, J., et al.: Research on data streams publishing of privacy preserving. In: IEEE International Conference on Information Theory & Information Security. IEEE (2015)
9. Ren, X., Yang, et al.: Research on CBK (L, K)-anonymity algorithm. Int. J. Adv. Comput. Technol. (2011)
10. Machanavajjhala, A., Gehrke, J., Kifer, D., et al.: ℓ-diversity: Privacy beyond k-anonymity. In: Proceedings of the 22nd International Conference on Data Engineering, ICDE 2006, 3–8 April 2006, Atlanta, GA, USA. IEEE (2006)
11. Sun, X., Wang, H., Li, J., et al.: (p+, α)-sensitive k-anonymity: a new enhanced privacy protection model. In: IEEE International Conference on Computer & Information Technology. IEEE (2014)
12. Tai-Yong, L.I., Chang-Jie, T., Jiang, W.U., et al.: k-anonymity via twice clustering for privacy preservation. J. Jilin Univ. **27**(02) (2009)

13. Chen, X., Wu, X., Wang, W., et al.: An improved initial cluster centers selection algorithm for K-means based on features correlative degree. Adv. Eng. Sci. **047**(001), 13–19 (2015)
14. Hong-Wei, L., Guo-Hua, L.: (L, K)-anonymity based on clustering. J. Yanshan Univ. (2007)
15. Chen, C.Y., Li, S.A., et al.: A clustering-based algorithm to extracting fuzzy rules for system modeling. Int. J. Adv. Comput. Technol. (2016)