International Journal of Security and Its Applications
Vol. 10, No. 4 (2016), pp.79-88
http://dx.doi.org/10.14257/ijsia.2016.10.4.09

Differential Privacy via Weighted Sampling Set Cover

Zhonglian Hu, Zhaobin Liu*, Yangyang Xu, Zhiyang Li

School of Information Science and Technology, Dalian Maritime University
No. 1, Linghai Road, Dalian, P.R.China, 116026
[1, 2, 3, 4]{huzhongnian, zhbliu, yyx.dlmu, lizy0205}@gmail.com*
**zhbliu@gmail.com*

Abstract

Differential privacy is a security guarantee model which widely used in privacy preserving data publishing, but the query result can't be used in data research directly, especially in high-dimensional datasets. To address this problem, we propose a dimensionality reduction method. The core idea of this method is using a series of low-dimensional datasets to reconstruct a high-dimensional dataset, it improves data availability eventually. The main issue of this method is the reconstruction integrity, so a special sampling via set cover model is proposed in this article, which builds a multidimensional composite marginal tables set as a new middleware in differential privacy model. As a result, any form of disjunctive queries can be answered, and the accuracy of data query is improved. The experiment results also show the effectiveness of our method in practice.

Keywords*: differential privacy, set cover, sampling*

1. Introduction

The weakness of data encryption and anonymization has been exposed gradually in recent years, not only in data publication, but also in network. The culprit of all those problems is that the background knowledge of attackers are hard to define [1]. Different privacy models have different assumptions of the background knowledge. With rigorous statistical definition, the differential privacy can provide one of the strongest privacy guarantees, which means even the most powerful attackers can't steal accurate personal information from published data.

The noise plays an important role in the differential privacy model. The possibility of deriving personal information from extra knowledge is fundamentally eliminated. But the data utility suffers from noise in multi-dimensional, especially in high dimensional disjunctive query. Because the query result will involve noise from a large number of entries. Dimensionality reduction is the most effective solution, and it is also one of the major issues in differential privacy research [2] [3].

After reducing dimension in datasets, it may causes serious data distortion. Because attributes are lost during the dimension reduction process. In order to improve the availability of query results and reduce the loss of attributes, we propose a new weighted sampling method based on the Set Cover Problem (SCP).

SCP is a traditional set learning problem. The mathematical model of this problem is widely used in facility location and workforce creation. Previous works on SCP are composed of three parts: 1) weighted set cover [4] [5], 2) maximum coverage [6] [7], 3) budgeted maximum coverage [8] [9]. Our problem is a weighted set cover problem. In section 3 we illustrate the details. Optimizing cost for fixed

* Corresponding author is Zhaobin Liu.

ISSN: 1738-9976 IJSIA

coverage is the main idea of weighted set cover problem. In our solution, we use this idea to find a group of marginal table with different dimension, which can reach the desired query coverage and minimize the noise in query result.

The remainder of this paper is organized as follows. Section 2 introduces existing analysis on differential privacy and set cover problem (SCP). Section 3 proposes our problem definitions formally, and section 4 describes our new sampling method and set cover model. Section 5 gives experiments, we apply our method in practice and compare it with existing methods.

2. Related Work

In this section, we introduce the recent research in differential privacy model and SCP. As mentioned earlier, the dimensionality reduction acts effectively in reducing noise in query result, and in our methods we come up with a SCP to solve the problem of attributes loss. Here we suggest the research status from two sides.

In the research of differential privacy, there has been works on data de-noising for high dimensional data set [10] [11]. Among them, a method which used a series of low-dimensional datasets to reconstruct a high-dimensional dataset is the most intuitive and effective, and it is based on marginal table. For example, there is a dataset about the hand habit in table 1. It involves three attributes, so we get the one 3-way marginal table in table 2, and two 2-way marginal tables in table 3. Now we add noises σ in each entire of table 2 and table 3 to ensure ϵ-differential privacy for publication. Here we want to count all the number of left-handed males Num_{left}. From table 2 $Num_{left} = 154 + 2\sigma$. In table 3 $Num_{left} = 154 + \sigma$. Obviously, the Num_{left} in table 3 gets smaller noise than in table 2.

Table 1. Hand Habits Records

ID	Sex	Parents	Hand Habits
1	Male	Left	Left
2	Female	Left	Right
3	Female	Right	Left
…	…	…	…
N	Male	Right	Right

Table 2. Three-Way Marginal Table

	000	001	010	011	100	101	110	111
Number	$100+\sigma$	$51+\sigma$	$25+\sigma$	$10+\sigma$	$73+\sigma$	$12+\sigma$	$81+\sigma$	$5+\sigma$

Table 3. Two-Way Marginal Table

	00	01	10	11
Number	$125+\sigma$	$61+\sigma$	$154+\sigma$	$17+\sigma$

Since this method proposed, many investigators have done research in this area. For example, Wahben *et al* came up with PriView method, which computed marginal tables for a number of strategically chosen sets of attributes that they call views, and then used these view marginal tables to reconstruct any desired k-way

marginal [12]. Micheal Hay *et al* carefully chooses a set of queries to evaluate, and then exploits consistency constraints that should hold over the noisy output [13]. Those methods used a group of marginal tables called views to cover a k-way marginal tables.

However, those views have the same dimensions, it's impossible to cover all kinds of statistical query. Such as in table 3, we can't use it to count the number of man with right-handed and his parents with left-handed. But in our view sets, the dimension of marginal tables is different, and we propose a sampling method to choose sets. Indeed, the weighted set cover model also help us to ensure the desired query coverage.

Weighted set cover problem is the algorithmic frameworks of our methods, also a classic NP-problem [14] [15]. The main issue of this problem is how to find the approximation subset sets in each iteration, a good subsets can reduce the time complexity of the algorithm. So in this paper, we propose a multistage stratified sampling algorithm to optimize the subsets.

3. Problem Statement

The basic approach of our method is organized as follows. Firstly, adding noise obeying the Laplace distribution Lap(1/ϵ) into each entries of original dataset, then we choose the marginal tables with different dimensions following the new sampling method, finally we use those marginal tables to reconstruct the middleware for publication.

So the input of our problem is a dataset which satisfies the ϵ-differential privacy as definition 1 [16].

Definition 1 (ϵ-differential privacy): There are two databases D_1 and D_2, differing in only one tuple, we call those neighboring datasets. Do any data query Q for all parts of neighbors D_1 and D_2. If all results S satisfy in the equation 1, then we say those databases satisfy ϵ-differential privacy.

$$\Pr\left[Q(D_1) = S\right] \le e^{\delta} \Pr[Q(D_2) = S] \tag{1}$$

Using the existing SCP algorithm to find the approximation marginal tables is hard and impossible. Because in an n-dimensional dataset, it will be $O(2^n)$ complexity when we traversed all marginal tables from 1 to m, $\sum_{i=1}^{m} C_i^1 = 2^n$. So here we divide the marginal tables into n-level, i-way marginal tables belonging to i-level. Then we choose the desired marginal table from each levels based on SCP. The problem can be defined as definition 2.

Definition 2 (Problem Definition): Given a ϵ-differential privacy dataset D, the dimension of dataset m, and a minimum query coverage fraction q. The marginal tables of D can be divided into m levels. The k-way marginal tables is involved in k-level. All marginal tables in the level have their own noise value, and we set it as the weight of marginal table n_i. We suggest finding some marginal table s_i from each levels. The number of all marginal tables is $\sum_{i=1}^{m} s_i = S$, and the subsets set of S marginal tables such that $\sum_{i=1}^{S} Cov(s_i) \ge q|D|$ and $\sum_{i=1}^{S} n_i$ is minimal.

Here $Cov(s_i)$ reflects the number of contrain attributes in marginal tables that has no included in S.

4. Algorithms

In this section, we propose our multistage stratified sampling algorithm based on weighted set cover.

4.1. Multi-stage Stratified Sampling Method

The main issue of weighted SCP is that we must traverse all possible subsets of the collection of attributes. The number of attributes equal to the dimension of datasets. For example, there is a collection of $B = \{a, b, c, d\}$. The elements in *B* is the attribute in dataset, so we say *B* is a 4 dimension dataset, and elements have their own weight *W*. We set *W* (*a*) = 1, *W* (*b*) = 4, *W* (*c*) = 10, and *W* (*d*) = 5. The weight of subset is the sum of each elements in it, *W* (*a*, *b*) = 1 + 4 =5. Now we set the coverage fraction *q* = 1. In order to find the solution subsets set, we must traverse all possible subsets of the combination and compare their weights. In this collection, we need traverse 16 times, so in an m-dimensional database it will be 2^m times. Obviously, it has the $O(2^m)$ time complexity, and has been proved to be a *NP-hard* problem [17].

To reduce the traversal times, we choose the desired subsets in each levels. So in the first step, we divided the marginal table of *m*-dimmensional dataset into *m* levels, marginal table is involved in levels according to their dimension, *i*-way marginal table in *i*-level. The number of subsets in *i*-level is C_m^i. We still have to choose s_i in each level and the time complexity is $O(m\sum C_m^1)$.

The coverage fraction of our problem is different, we not only need that elements of solution subsets set contains all the attributes, we but also need that *S* can represent all boolean conjunctions, it means *q* = 1. Our requirements are much difficult to satisfy, but the *k*-way marginal table has a special feature, that is a *k*-way marginal table can answer all disjunctive query in *k*-1 way marginal table, when $s_{i-1} \subset s_i$. *k*-2 to 1 way marginal table also can be answered with the same condition, but we ignore it because, it will mix much more noises.

With this feature, we optimize our Multi-stage Stratified Sampling Method (MSSM). Traversing all marginal tables in each level is unnecessary, because the disjunctive query which can not be answered in *k*-way marginal table, can be answered the corresponding *k* + 1 marginal table. So in the second step, we just need traverse parts of the marginal tables. In the 1-level, we still have to traverse all the subsets, but the C_m^1 times is acceptable. We only need to choose c_1 subsets, which can get in equation 2. After that the sampling in the *k* (*k*>1) level should comply with three conditions, 1) covering all the non-covered attributes in *k*-1 level, if there is more than 1 attributes, covering all the subsets that involve both, 2) the marginal tables which contain the subsets in k-1 level are not be sampled, the number of sampling number in *k* level is uncertain, but we can get a range of it. From these two conditions, we can get that number from two parts p_1 and p_1, the range of p_1 is in equation 3, the $p_2 = C_{m-c_{k-1}-n}^k$. So the $c_k = \max(p_1, p_2)$. 3) most importantly, the universal set must be included in, which is the guarantee of the efficiency of our algorithm, and we can use it to answer parts of the disjunctive query in *m*-1 level.

$$C_n^2 - \sum_{i=n-1}^{n-c_1} C_i^1 \le \left\lceil \frac{C_n^2}{2} \right\rceil \tag{2}$$

$$C_{n-m}^{i-1} < p_1 < C_{n-m}^{i} \tag{3}$$

The detail steps of this method is shown in Algorithm 1. Taking the collection of *B* that mentions in previous for example. The dimension of *B* is 4, so we divide the marginal tables in *B* to 4 levels. First of all, we choose the 1-way marginal tables in one-level. We can get 4 marginal tables in one level. According to our algorithm, we choose at least one marginal table, here we choose *a*. In respect of sampling condition, we can't choose marginal table which contains the element *a*, so we just need to choose in only 3 entries (*bc, bd, cd*), here we choose *bd* and *cd*. In the three level, we have to choose the marginal tables which contain *a* to equalize the loss of query coverage, and it's not contains elements that in two level, so we choose *abc*. In the four level, *abcd* is chosen.

In addition, we suggest to use a B-tree as the storage data structure of sampling items in Figure 1. B-tree is a self-balancing tree data structure that keeps data sorted and allows searches, sequential access, insertions, and deletions in logarithmic time [18]. There is two reasons why we choose this B-tree to save sampling marginal tables. 1) According to our weighted set cover model in section 4.2, we usually choose the marginal tables with high $Cov(s)$, B-tree is multiple search tree and it will make traverse conveniently. Moreover, the feature of self-balancing do helpful in pruning our algorithm. 2) Memory optimization. The space occupancy is the short coming of B-tree, but we can build the B-tree in the disk, then let the frequent items stored in memory. By this way, we can save a certain time, and improve the usage of memory.

4.2. Weighted Set Cover Model

The sampling method just keeps the query coverage. We need to reduce the noise in each subsets. And we also wish to cover more uncovered attributes when sampling in each level. So we sort the marginal tables which is chosen in each level based on their noise, and then we sort it again based on their $Cov(s_i)$. Then we choose the marginal table with high value of $Cov(s_i)$ and low value of noise.

4.3. Consistency Processing

In a same level, two marginal tables usually contain some identical attributes. When users query in the publication middleware, they can get the query result of same attributes from both of them. But with the range noise obeying Laplace distribution, the answer of the same query may be different.

To address this problem, we use the consistency algorithm proposed in [19]. But in our environment, we change the working condition. The algorithm uses the average of those values as the query result. But our way of getting query result is different. When user use the collection marginal tables to query, first it finds the result in the corresponding dimension marginal tables. If you do a 2-way disjunctive query, it finds the result in 2-way marginal tables. If it cannot get the result, then it tries to find the result in the next dimension marginal tables, such as 3-way. So, we just use this method between the marginal tables with the same dimension.

Algorithm 1 Multi-stage Stratified Sampling Method

Input: a database with ϵ-differential D, $|D| = m$, query coverage fraction q, i levels

1: $S = \varnothing$

2: **while** $q < 1$ or $i <= m$ **do**

3: **if** $i = 1$ **then**

4: choose *s* 1-way marginal tables, $C_{m-s}^2 < C_m^2 / 2$

5: **else if** $1 < i < m$ **then**

6: choose *s* k-way marginal tables, $s = \max(p_1, p_2)$

7: $S = S \cup s$, $i++$

8: put m-way marginal table into S

9: **return** S

5. Experiment

In this section, we compare our algorithm with PriView algorithm used in data utility and time complexity, because PriView is widely used. The experiment results can show the efficiency and accuracy of our algorithm.

5.1. Introduction of Data Set

We use a data set from UCI Machine Learning Repository, called Online Video Characteristics and Transcoding Time Dataset Data Sets in our experimental comparison. This dataset is composed of two tsv files named 'youtube_video.tsv' and 'transcoding_measure.tsv'. We only use the first file in our experiment. The first files contains 10 columns of fundamental video characteristics for 1.6 million youtube videos. It contains YouTube video id, duration, bitrate (total in Kbits), bitrate (video bitrate in Kbits), height (in pixle), width (in pixles), framrate, estimated framerate, codec, category, and direct video link [20].

5.3. Comparison on Data Utility

First, we compare with PriView in data utility. The error of real dataset and noise dataset query results can reflect the data utility, but errors in each query result are different, so we use the mean absolute difference (MAD) to reflect the global errors in a publication noise middleware.

The steps of our experiment are as follow. 1) We get the publication noise middleware from our method and PriView on each dimension. The dimension of dataset is 10, so we can get 10 kinds of middleware. 2) Choosing 5 kinds of disjunctive query, we get those query results in each middleware and evaluate values of MAD.

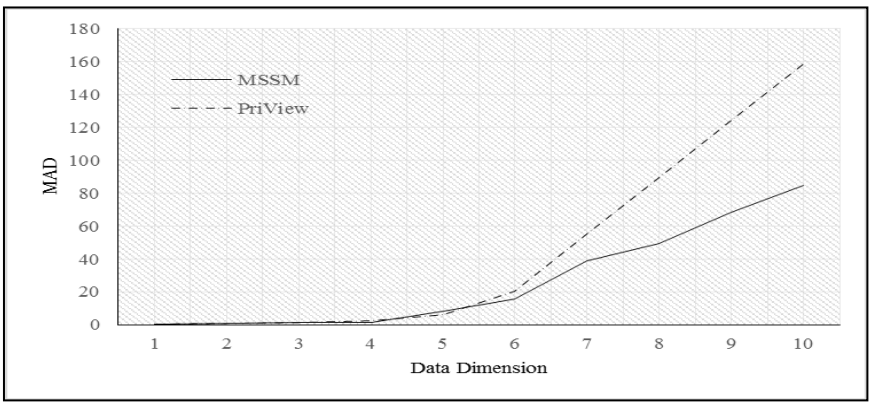

Figure 1. The MAD distribution of MSSM and PriView

Figure 1 plots the MAD of the MSSM and PriView as a function of data dimension. The data dimension is from 1 to 10. For low dimension, the MAD is similar, because the optional set is small in 1 to 3 dimension. So the middleware in MSSM and PriView are usually the same. But for high dimension, the MAD of MSSM is significantly less than PriView, since PriView needs some low dimension disjunctive query in high way marginal tables, which means more noise is included.

5.2. Comparison on Time Complexity

Second, we compare with PriView in time complexity. The experiment process is similar with the previous one. But we record the running time and the number of tuples in middleware.

Figure 2 explains why our MSSM algorithm can reduce the time complexity. With the increasing of dimension, the number of optional sets is larger, so the running time of PriView grows fast with the data dimension. But our method performs better than it.

In Figure 3, we show the size of tuples in middleware. It may influence the time of consistency processing in data query. The smaller the size is, the better users' experiences are. Our method performs better than PriView.

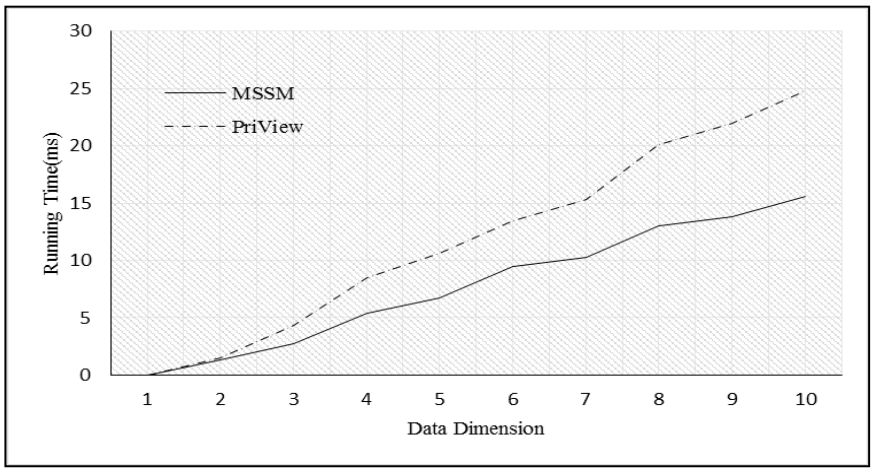

Figure 2. The Running Time Distribution of MSSM and PriView

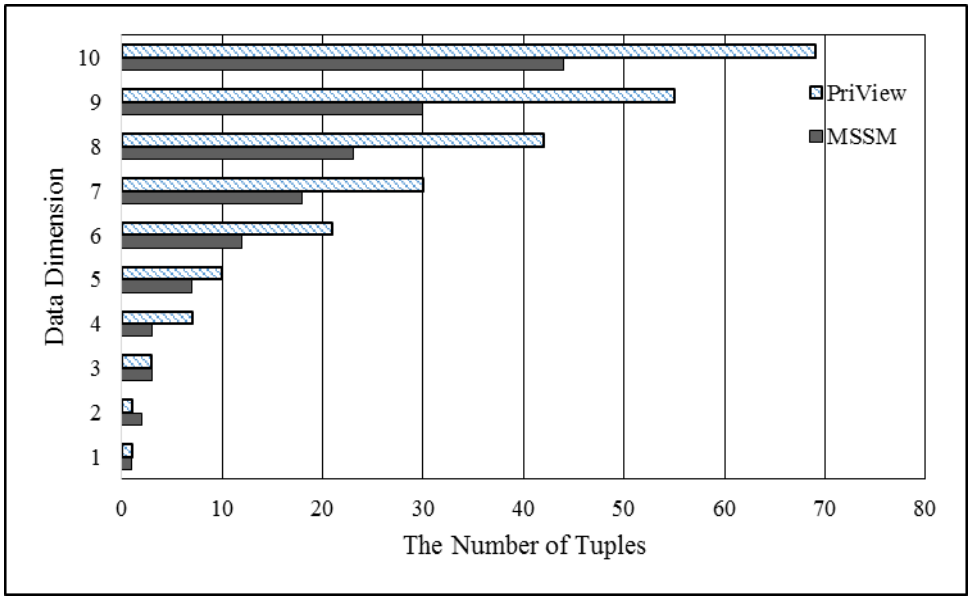

Figure 3. The Number of Tuples Distribution of MSSM and PriView

6. Conclusion

We proposed efficient sampling method MSSM based on weighted SCP for Differential Privacy. Traditional algorithms can not get accurate query results, and have low query coverage. To address this problem, we suggest the MSSM sampling method, and apply it to the differential privacy model. Results from the experiment show that our method performs better in both data utility and time complexity than exist method.

Acknowledgements

This work is supported by the National Science Foundation for Distinguished Young Scholars of China under grant no. of 61225010, NSFC under grant nos. of 61370198, 61370199 and 61300187, The Fundamental Research Funds for the Central Universities under grant nos. of 3132014215 and 3132014208.

References

[1] D Kifer . “Attacks on privacy and deFinetti's theorem”. Proceedings of ACM SIGMOD International Conference on Management of data. **(2009)** June 29-July 2； Rhode Island, USA

[2] S Peng, Y Yang, Z Zhang, M Winslett, Y Yu. DP-tree: indexing multi-dimensional data under differential privacy. Proceedings of ACM SIGMOD International Conference on Management of data. **(2012)** May 20-24: Scottsdale, AZ, US

[3] S E. Fienberg, A Rinaldo, X Yang. “Differential Privacy and the Risk-Utility Tradeoff for Multi-dimensional Contingency Tables”. Privacy in Statistical Databases - UNESCO Chair in Data Privacy, International Conference **(2010)** September 22-24: Corfu, Greece.

[4] S Stergiou, K Tsioutsiouliklis. “Set Cover at Web Scale. Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining”. **(2015)** August 10-13: Sydney, NSW, Australia

[5] S Har-Peled, M Lee. “Weighted geometric set cover problems revisited”. Journal of Computational Geometry. vol. 3, no. 1, **(2012)** pp. 65-85 **.**

[6] D Kilinç, M Ozger, Ö B. Akan. “On the Maximum Coverage Area of Wireless Networked Control Systems With Maximum Cost-Efficiency Under Convergence Constraint”. IEEE Trans. Automat. Contr. , **(2015) ,** pp. 1910-1914.

[7] A Al-Hourani, S Kandeepan, S Lardner. “Optimal LAP Altitude for Maximum Coverage”. Wireless Communications Letters, IEEE. vol. 3, no. 6, **(2014)**, pp. 569-572.
[8] B Caskurlu, V Mkrtchyan, O Parekh, K. Subramani. “On Partial Vertex Cover and Budgeted Maximum Coverage Problems in Bipartite Graphs”. Theoretical Computer Science - 8th IFIP TC 1/WG 2.2 International Conference. **(2014)** September 1-3 : Roma, Italy .
[9] F Cheng, Z Zhu, S Li, H Cheng, F Wan. “An SQP Algorithm for Optimization with Linear Complementary Constraints”. Journal of Harbin University of Science and Technology **(2014) .**
[10] K Mivule, C Turner, “Applying Moving Average Filtering for Non-interactive Differential Privacy Settings. Procedia Computer Science”. no. 36, **(2014)**, pp. 409-415.
[11] G Barthe, M. Gaboardi, G Arias E J, et al. “Proving differential privacy in Hoare logic”. Computer Security Foundations Symposium (CSF). **(2014)** July 19-22: Vienna, Austria.
[12] W H. Qardaji, W Yang, N Li. “PriView: practical differentially private release of marginal contingency tables”. Proceedings of ACM SIGMOD International Conference on Management of data **(2014)** June 22-27: Snowbird, UT, USA.
[13] M Hay, V Rastogi, G Miklau, D Suciu. “Boosting the Accuracy of Differentially Private Histograms through Consistency”. PVLDB, vol. 3, no. 1, **(2010)**, pp. 1021-1032.
[14] O Aichholzer, W Mulzer, A Pilz. “Flip Distance Between Triangulations of a Simple Polygon is NP-Complete”. Discrete & Computational Geometry, vol. 54, no. 2, **(2015)**, pp. 368-389 .
[15] R de Haan, S Szeider, “The parameterized complexity of reasoning problems beyond NP”. arXiv preprint arXiv: **(2013)**, pp. 1312-1672.
[16] C Dwork. “Differential Privacy. Automata, Languages and Programming”, 33rd International Colloquium. **(2006)** July 10-14: Venice, Italy.
[17] W Li. “New Variants of Lattice Problems and Their NP-Hardness”. Information Security Practice and Experience - 10th International Conference. **(2014)** May 5-8. Fuzhou, China.
[18] B-tree: https://en.wikipedia.org/wiki/B-tree.
[19] E Lehrer, D Samet. “Belief consistency and trade consistency”. Games and Economic Behavior, no. 83, **(2014)**, pp. 165-177.
[20] Online Video Characteristics and Transcoding Time Dataset http://archive.ics.uci.edu/ml/datasets/

Authors

Zhonglian Hu, The main research interests of him are differential privacy and set theory. In 2010, he received the bachelor's degree from the Dalian Maritime University, China. At present, he is studying for master's degree at the School of Information Science and Technology, Dalian Maritime University.

ZhaoBin Liu, He is a Professor in School of Information Science and Technology, Dalian Maritime University, China. He received his Ph.D. in Computer Science from Huazhong University of Science and Technology in China in 2004. His research areas include Cloud computing, Big data, Computer Networks and Embedded Systems. He has more than 60 publications in international journals, conference proceedings as well as book chapters, and has successfully coordinated several research projects funded by various funding agencies across China.

Yangyang Xu, He received the bachelor's degree from the Dalian Maritime University, China, in 2010. Currently, he is working toward the master's degree at the School of Information Science and Technology, Dalian Maritime University. His main research interests include data privacy protection and data mining.