Privacy Preservation of Stream Data Patterns Using Offset and Trusted Third Party Computation in Retail-Shop Market Basket Analysis

Keshavamurthy B.N. and Durga Toshniwal

Department of Electronics & Compuetr Engineering
Indian Institute of Technology Roorkee,
Uttarakhand, India
{kesavdec,durgafec}@iitr.ernet.in

Abstract. Privacy preservation is widely talked in recent years, which prevents the disclosure of sensitive information during the knowledge discovery. There are many applications of distributed scenario which includes retail shops, where the stream of digital data is collected from time to time. The collaborating parties are generally interested in finding global patterns for their mutual benefits. There are few proposals which address these issues, but in the existing methods, global pattern computation is carried out by one of the source itself and uses one offset to perturb the personal data which fails in many situations such as all the patterns are not initiated at the initial participating party. Our novel approach addresses these problems for retail shops in strategic way by considering the different offsets to perturb the sensitive information and trusted third party to ensure global pattern computation.

Keywords: Privacy preservation, steam data, offset computation, trusted third party.

1 Introduction

In recent years, due to the advancement of computing and storage technology, digital data can be easily collected. It is very difficult to analyze the entire data manually. Thus a lot of work is going on for mining and analyzing such data.

In real many world applications which include retail-shops data is distributed across different sources. The distributed data base is comprised of horizontal, vertical or arbitrary fragments. In case of horizontal fragmentation, each site has the complete information on a distinct set of entities. An integrated dataset consists of the union of these datasets. In case of vertical fragments each site has partial information on the same set of entities. An integrated dataset would be produced by joining the data from the sites. Arbitrary fragmentation is a hybrid of previous two.

The key goal for privacy preserving data mining is to allow computation of aggregate statistics over an entire data set without compromising the privacy of private data of the participating data sources.

The key techniques in privacy preservation to perturb the sensitive data, includes randomization. The randomization method is a technique in which for privacy

N. Meghanathan et al. (Eds.): CCSIT 2011, Part III, CCIS 133, pp. 358–366, 2011.

preserving data mining, offset or noise is added to the sensitive data in order to mask the attribute values of records. The offset added is sufficiently large so that individual record values cannot be recovered.

Most of the methods for privacy computation use some transformation on the data in order to perform the privacy preservation. One of the methods widely used in distributed computing environment for a computation of global pattern is secure multi party computation.

The reminder of this paper is organized as follows: section 2 gives a formal definition of the problem definition of this paper and discusses the randomization and secure multiparty computation on which the proposed work is applied. In section 3, we presented proposed module for mining the stream data of retail-shop by using trusted third party and different offsets. In section 4, includes performance evaluation. We conclude our work in section 5.

2 Preliminaries

2.1 Problem Statement

A lot of research papers have discussed the privacy preserving mining across distributed databases. Major drawbacks with the existing techniques are that the global pattern computation is done at one of the data source itself which violates the privacy concern majorly. The proposed work address this issues very efficiently by using a trusted third party to alleviate privacy preservation across distributed data sources. Secondly to perturb the sensitive items uses one offset value which fails in many practical scenario of retail-market such as all the items need not be sold at a particular shop. This technical gap is resolved by taking two offsets, one which will be used for continuing items and the later used for newly initiated items at each data source to perturb the sensitive items of collaborating parties.

2.2 Background

Randomization Method

The most widely used technique in privacy preservation to perturb the sensitive attributes is randomization technique. It is described as follows: Consider a set of data denoted by $X = \{ \mathrm{x}_1, \mathrm{x}_2, ...\mathrm{x}_\mathrm{N} \}$. For record $x_i \in x$, we add offset component which is drawn from the probability $\mathrm{f}_\mathrm{y}(\mathrm{y})$. These offset components are drawn independently, and are denoted $\mathrm{y}_1, \mathrm{y}_2, ...\mathrm{y}_\mathrm{N}$ thus, the new set of distorted records are denoted by $x_1 + y_1, ..., x_n + y_N$. We denote this new set of records by $z_1, ..., z_N$. In general, it is assumed that the variance of the added offset is large enough, so that the original record values cannot be easily guessed from the distorted data. Thus, the original records cannot be recovered, but the distribution of the original records can be recovered Thus, if x be the random variable denoted the data distribution for the original record, y is the random variable describing the offset distribution, and z be the random variable denoting the final record, we have

$$Z=X+Y. \tag{1}$$

$$X=Z-Y. \tag{2}$$

Secure Multiparty Computation

A number of technologies have recently been proposed in order to perform the data mining tasks in a privacy preserving way. The most promising technique which is widely used to alleviate privacy preservation concerns in distributed database scenario includes secure multi party computation.

Generally in a distributed database environment to compute the global patterns of n data sources. In secure multiparty computation, when there are n data sources $DS_0, DS_1, ..., D_{n-1}$ such that each

In case of distributed environment, the most widely used technique in privacy preservation mining is secure sum computation [16]. Here when there are n data sources $DS_0, DS_1, ... D_{n-1}$ such that each DS_i has a private data item $d_i, i = 0,1,...,n-1$ the parties want to compute $\sum_{i=0}^{n-1} d_i$ privately, without revealing their private data d_i to each other. The following method was presented:

We assume that $\sum_{i=0}^{n-1} d_i$ is in the range $[0, m-1]$ and DS_j is the protocol initiator. Following steps were followed:

1. At the beginning DS_j chooses a uniform random number r within $[0, m-1]$.
2. Then DS_j sends the sum $d_i + r \pmod{m}$ to the data source $DS_j + 1 \pmod{n}$
3. Each remaining data sources DS_i do the following: upon receiving a value x the data source DS_i sends the sum $d_i + x \pmod{m}$ to the data source $DS_i + 1 \pmod{n}$.
4. Finally, when party DS_j receives a value from the data source $DS_{n-1} - 1 \pmod{n}$), it will be equal to the total sum $r + \sum_{i=0}^{n-1} d_i$. Since r is only known to DS_j it can find the sum $\sum_{i=0}^{n-1} d_i$ and distribute to collaborating parties.

3 Proposed Work

Our proposed work the modify the existing secure multiparty computation by introducing trusted third to enhance privacy preservation and also uses two offset values to perturb the sensitive attribute values.

3.1 Modified Secure Multiparty Computation with Two Offset Computation

Assumption

Here there are n data sources $DS_0, DS_1, ...D_{n-1}$ such that each has a private data item d_i, $i = 0,1,...n-1$, the parties want to compute $\sum_{i=0}^{n-1} d_i$ privately, without revealing their private data d_i to each other. The following method was presented: We also assume that $\sum_{i=0}^{n-1} d_i$ is in the range $[0, m-1]$ and trusted third party is the protocol initiator.

Procedure

At Trusted Third Party

1. The collaborating parties interested in global pattern computation are connecting to the trusted third party.
2. Trusted third party choose a uniform random r to select one of the N parties as an initiator (i.e., selects one of the DS_i, $i = 0,1,...,n-1$) and send the random offset to that party DS_i.
3. Trusted third party sends two offsets oldoffset and newoffset for each collaborating party except initiator. Initiator receives only newoffset.
4. Finally, when trusted third party receives a value from the data source $DS_{j-1} - 1 \pmod{n}$, it will be equal to the total sum $r + \sum_{i=0}^{n-1} d_i$ Since r is only know to trusted third party, it can find the sum $\sum_{i=0}^{n-1} d_i$ and distribute the same to collaborating data sources.

At Trusted Third Party

Collaborating party

1. If he is an initial collaborator then add its own data value D_i to newoffset value received from trusted third party.
2. For each items at DS_i following action taken at collaborator side except the last collaborator:

   a. If there are no new items in compare with the item list obtained from D_{i-1}, then sum $D_i + \text{oldoffset}$.

   Else

b. For all the new items in compare item list obtained from D_{i-1} then sum $d_i + newoffset$ at D_i.

c. Send the final list of items to the next collaborating party specified by trusted third party

3. The last collaborating party performs

a. If there are no new items in compare with the item list obtained from D_{i-1} then sum $d_i + oldoffset$.

Else

b. For all the new items in compare item list obtained from D_{i-1} then sum $d_i + newoffset$ at D_i.

c. Send the final list of items to the party trusted third party.

4 Analysis and Evaluation

This section mainly discusses the execution and performances issues of the algorithm presented in the previous section. In section A, we introduce the practical problem we considered. Section B discusses the analysis part of the algorithm.

4.1 Evaluation Environment

Experiments were conducted on the following scenario: Trusted third party with 10 data sources, each data source contains 3 records. Basically there are two tables which operate by each collaborating party, one local database table, to keep local items information and the other sendtable, to have complete information of all the items till that party.

The following table's of Fig.1 gives the complete information of first three collaborating parties followed by the analysis of the data at different data sources and trusted third party. Initially all the parties interested in global pattern computation makes a request to trusted third party. Trusted third party select one them as initiator and send random offset number i.e., offset-1 to perturb his sensitive data. At party1, adds his data with offset1 send by trusted party then result will be send to the next party which is specified by trusted third party called as party2. At party2 if there are new items in compare with the existing list which is received from party1 then he has to add offset-1 to them for rest, add offset-2 to preserve the privacy of sensitive data and send the result to next logical party which is specified by trusted third party and this will continue till the last party. At last party he adds his data with offset-1 to new items which initiated from him, for the rest add offset-2 then the final resulting table will be sent to the trusted third party for the global pattern computation.

At trusted third party he subtracts the random value from aggregate values obtained from last party. Then at third party the mining operation for the aggregate values of the collaborating party will be carried out and the mined result will be send back to the collaborating parties. In the entire process the trusted party will never gets details of the items of any of the collaborating party individually and the collaborating parties are the constituents of distributed database scenario so they can only know the party who is adjacent to him but he never knows the complete details such as what is his order which may helps in getting to know how many people are before and after him so that he can misuse the data which is receive in the course.

Table 1. Party1 local data

Item	Item Frequency	Offset-1	Perturbed Value
A	10	50	60
BC	15		65
DEF	20		70

Table 2. Party1 send data

Item	Perturbed Value
A	60
BC	65
DEF	70

Table 3. Party2 local data

Item	Item Frequency	Offset-1	Offset-2	Perturbed Value
AB	15	50	120	135
BC	5			75
DEF	75			145

Table 4. Party2 send data

Item	Perturbed Value
A	130
AB	135
BC	140
DEF	215

Table 5. Party3 local data

Item	Item Frequency	Offset-1	Offset-2	Perturbed Value
BC	10	100	220	110
A	15			115
FG	35			255

Table 6. Party3 send data

Item	Perturbed Value
A	245
AB	235
BC	250
DEF	335
FG	255

Table 7. Party10 local data

Item	Item Frequency	Offset-1	Offset-2	Perturbed Value
A	30	100	560	130
BC	20			120
DEF	25			125

Table 8. Party10 send data

Item	Perturbed Value
A	756
AB	615
AC	645
B	615
BC	645
D	670
DEF	786
FG	595
G	595
H	620

Fig. 1. Data perturbation at collaborating parties (Party1to Party 10)

4.2 Analysis of Algorithm

Table 9 gives the data items of 10 distributed data sources such as party1 to party10 along with the item frequency. The corresponding feature graph for the input data items are given in Fig.2. The integrated data of all the data sources party1 to party10 at trusted third party after deducting the integrated offset supplied to the different collaborating parties will be given by table10 and the corresponding feature graph is drawn in Fig.2. The feature graph we obtained for at the integrated data (Table 8) at the source side is same as the feature graph which we drawn at trusted party after offset subtraction at Fig. 3(Table 10).

Table 9. Items at different data sources

Party| Item	Party1	Party2	Party3	Party4	Party5	Party6	Party7	Party8	Party9	Party10
A	10	0	15	50	100	0	0	0	30	0
B	0	0	0	0	0	55	0	0	0	150
AB	0	15	0	0	0	0	30	10	0	0
Ac	0	0	0	0	0	0	20	55	0	0
BC	15	5	10	15	20	0	0	0	20	0
D	0	0	0	0	0	0	40	70	0	0
DEF	20	75	0	0	35	70	0	0	25	0
FG	0	0	0	0	0	0	0	0	0	70
G	0	0	0	35	0	0	0	0	0	0
H	0	0	0	0	0	60	0	0	0	5

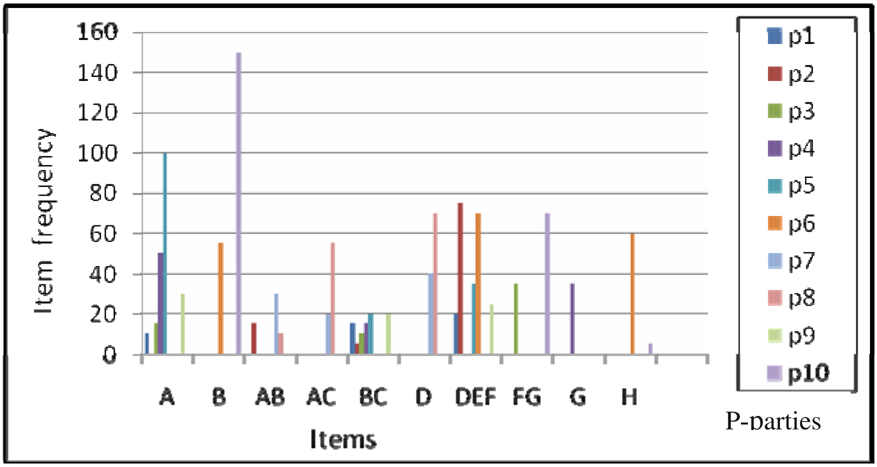

Fig. 2. Support of items at different data sources or collaborating parties

Table 10. Integrated items for collaborating parties at trusted third party

Items	Frequency
A	205
B	205
AB	55
AC	75
BC	85
D	110
DEF	225
FG	70
G	35
H	65

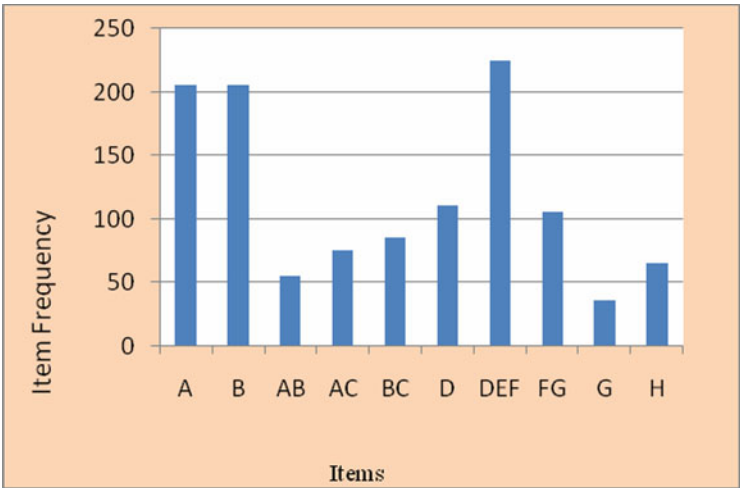

Fig. 3. Global patterns of items at trusted third party

5 Conclusion

The proposed work well suits with synthetic data and also demonstrated the efficient way of privacy preservation over distributed scenario of the retail-shop market basket analysis. The worst possibility of information misuse can take place at the second random party who can get to know the only first party details but which is always negligible in the entire crowd. The proposed model is scalable and can be used for real data set in future.

References

1. Huang, J.-W., Tseng, C.-Y., Ou, J.-C., Chen, M.-S.: A General Model for Sequential Pattern Mining with a Progressive Database. International Journal of Knowledge and Data Engineering 20, 1153–1167 (2008)

2. Mhatre, A., Verma, M., Toshniwal, D.: Privacy Preserving Sequential Pattern Mining in Progressive Databases using Noisy Data. In: International Conference on Information Visualization, pp. 456–460 (2009)
3. Agrawal, R., Srikant, R.: Privacy-preserving data mining. In: ACM SIGMOD Conference on Management of Data, New York, USA, pp. 439–450 (2000)
4. Samarati, P.: Protecting Respondents' Identities in Micro data Release. IEEE Trans. Knowledge Data Eng. 13(6), 1010–1027 (2001)
5. Fung, B., Wang, K., Yu, P.: Top-Down Specialization for Information and Privacy Preservation. In: ICDE Conference, pp. 205–216 (2005)
6. LeFevre, K., De Witt, D., Ramakrishnan, R.: Incognito: Full Domain K-Anonymity. In: ACM SIGMOD Conference, Maryland, pp. 49–60 (2005)
7. LeFevre, K., DeWitt, D., Ramakrishnan, R.: Mondrian Multidimensional K-Anonymity. In: ICDE Conference, pp. 25–25 (2006)
8. Park, H., Shim, K.: Approximate Algorithms for K-anonymity. In: ACM SIGMOD Conference, Beijing, China, pp. 67–78 (2007)
9. Wang, K., Yu, P., Chakraborty, S.: Bottom-Up Generalization: A Data Mining Solution to Privacy Protection. In: ICDM Conference, pp. 249–256 (2004)