MINING ASSOCIATION RULE FOR HORIZONTALLY PARTITIONED DATABASES USING CK SECURE SUM TECHNIQUE

Jayanti Danasana[1], Raghvendra Kumar[1] and Debadutta Dey[1]

[1] School of Computer Engineering, KIIT University, Bhubaneswar, ODISHA, INDIA
jayantifcs@kiit.ac.in, raghvendraagrawal7@gmail.com and
dev.dey009@gmail.com

ABSTRACT

The security of the large database becomes a serious issue while sharing the data to the network against unauthorised access. However in order to provide the security many researchers cited the issue of Secured Multiparty Computation (SMC) that allows multiple parties to compute some function of their inputs without disclosing the actual input to one another. Secure sum computation method is popularly and widely accepted due to its simple and thorough solution. The outcomes of our proposed procedure provide a significant result so that it becomes impossible for semi honest party to know the private data of some other sites.

KEYWORDS

Secure Multiparty Computation, trusted third site, horizontal partition, computational complexity, Ck secure sum protocol.

1. INTRODUCTION

The developments of computed technology in last few decades are used to handle large scale data that includes large transaction financial data, bulletins, emails etc. Hence information has become a power that made possible for user to voice their opinions and interact. As a result revolves around the practice, data mining [7] come into sites. Association rule mining is one of the Data Mining techniques used in distributed database. In distributed database the data may be partitioned into fragments and each fragment is assigned to one site. The issue of privacy arises when the data is distributed among multiple sites and no other party wishes to provide their private data to their sites but their main goal is to know the global result obtained by the mining process. However privacy preserving data mining came into the picture. As the database is distributed, different users can access it without interfering with one another. In distributed environment, database is partitioned into disjoint fragments and each site consists of only one fragment. Data can be partitioned in three different ways, that is, horizontal partitioning, vertical partitioning and mixed partitioning shows in figure no. 1. Again the details are discussed.

1.1 Partitioning of Database

Data can be partitioned in three different ways that is, like horizontally partitioned data, vertically partitioned data or mixed partitioned data.

Horizontal partitioning: - The data can be partitioned horizontally where each fragment consists of a subset of the records of relation R. Horizontal partitioning [3] [9] [10] [11] divides a table into several tables. The tables have been partitioned in such a way that query references

DOI : 10.5121/ijdps.2012.3613

are done by using least number of tables else excessive UNION queries are used to merge the tables sensibly at query time that can affect the performance.

Vertical partitioning: - The data can be divided into a set of small physical files each having the subset of the original relation, the relation is the database transaction that normally requires the subsets of the attributes.

Mixed partitioning: - The data is first partitioned horizontally and each partitioned fragment is further partitioned into vertical fragments and vice versa.

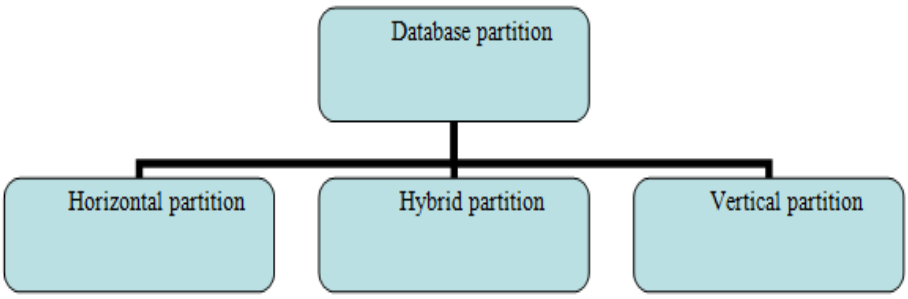


Figure 1. Different types of partition of Database

The idea is to build up a well organised method that enables a secure computation along with minimizing the amount of private data that each party discloses to other. Privacy preserving association rule mining may be used to solve these problems for horizontally partitioned database.

The rest of the paper is organised as follows. Section 2 describes the various works that related to association rule mining, distributed association rule mining and secure multiparty computation. Section 3 describes about the Ck secure sum protocol in horizontal partitioned database. Section 4 gave an experimental implementation and analysis of proposed method.

2. RELATED WORKS

The market basket analysis used association rule mining [3][6] in distributed environment. Association rule mining [1][2][7] is used to find rules that will predict the occurrence of an item and based on the occurrences of other items in the transaction, search patterns gave association rules where the support will be counted as the fraction of transaction that contains an item X and an item Y and confidence can be measured in a transaction the item i appear in transaction that also contains an item X

Support (s): - Fraction of transactions that contain both X and Y

Support (X->Y) = P (X∪Y)/T

Confidence(c): - Measure show often items in Y appear in transactions that contain X.

Confidence (X->Y) = Support (X∪Y) / Support (X)

Privacy preserving distributed mining of association rule [6][7] for a horizontally partitioned dataset across multiple sites are computed as follows where I = $\{i_1, i_2, .i_n\}$ be a set of items and T = $\{T_1, T_2, \ldots T_n\}$ be a set of transactions where each $T_i \subseteq I$. A transaction T_i contains an item

set $X \subseteq I$ only if $X \subseteq T_i$. An association rule implication is of the form $X \Rightarrow Y (X \cap Y = 0)$ with support S and confidence C if S% of the transactions in T contains $X \cup Y$ and C% of transactions that contain X also contain Y. In a horizontally partitioned database, the transactions are distributed among n sites.

Support $(X \Rightarrow Y)$ = probe $(X \cup Y)$ /Total number of transaction

The global support count of an item set is the sum of all local support counts.

Support g (X) = Support1(x) +Support2(x) +……+Support n(x).

Confidence of rule $(X \Rightarrow Y)$ = Support $(X \cup Y)$ / Support(X)

The global confidence [8] of a rule can be expressed in terms of the global support.

Confidence g $(X \Rightarrow Y)$ = Support g $(X \cup Y)$ / Support g(X)

The basis of this algorithm [6][7] is the apriori algorithm that uses K-1 frequent sets. The problem of generation size of one item set may be carried out with secure computation on multiple sites by generating the candidate set, the pruning method, finding the union of large item set

2.1. Secure Multi Party Computation (SMC)

The concept of Secure Multiparty Computation (SMC) [6] started when the Yao proposed the millionaire's problem in which two parties wanted to know who richer person was without disclosing the individual information regarding each other parties. In secure sum protocol basically divide the whole database into number of different parties and each site have their particular sequence number (s_1, s_2,……s_n) so that if s_2 want to the value of data of s_1 and s_3 then site s_2 will never able to know the data of other parties so in Ck secure sum very useful to provide a security to database with data leakage is zero percent. In Ck secure sum protocol the position of every site changes every time after computation of the first if there are four parties then number of round is three (if there are n number of parties then numbers of round is n-1). In horizontal partitioning of data, data is distributed among sites where the number of site will be greater than two. And no site is considered as a trusted party. All the party have their individual private data and no other party will able to know other party data. SMC problem can be extensive and expanded by Goldreich, Michalli and Wigderson [12] and others [13]. It has been proved that for any function there is a secure multiparty solution [14].

3. PROPOSED WORK

Assumption for the proposed work are taken as the database is horizontally partitioned and distributed among sites and the total number of sites is greater than two. The sites are considered as trusted site and all the site contain their own private data and no other site will be able to know other site data .In this method, basically, hash based secure sum technique [7] has been used. In secure sum each site will determine their own data value and send to predecessor site that near to original site and this goes on till the original site collects all the value of data after that the parent site will determine the global support and global confidence [6] [10] and it also not necessary that the result found is globally frequent or infrequent depending on value which will create after collecting all the value.

We have considered four sites s_1, s_2, s_3, s_4 where the sites are interchanging its position with another by following the algorithm. The secure sum protocol [9] is based on changing neighbours in each round of segment computation. The number of the site s_1 is selected as the protocol initiator site which starts the computation by distributing the first data segment. The site traverses towards s_n in each round of the computation. The number of parties for this protocol must be four or more. When all the rounds of segments summation are completed the sum is announced by the protocol initiator site. The steps are as follows.

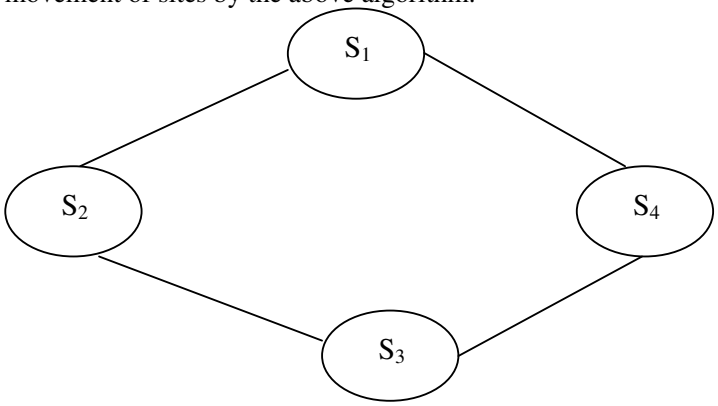
Algorithm:

Step1: Each site will determine their frequent item sets and infrequent item sets and store the data value on memory.
Step2: Each site will generate their own random number because we are using hash based secure sum protocol so that each site have two random number one of its own and other is received by previous site.
Step3: Now the site1 will determine the partial support value by using the following formula.
$Ps_j = X_j$ .support – Min support*|DB|-RN1-RNn where RN is random number.
After that site1 determine the mask value
$PS_j = PS_j$ + mask value
Step4: site2 compute the PS_j For each item received the list using the formula
$PS_j = PS_j + X_j$. Sup - min*support |DB| +RN1-RN (i-1)
Step5: After that the value of PS_j determined by site2 send to next coming site and after that all the value is send to the original Site and that original site will determine all global support.
Step6: site1 will find whether that global support is greater than zero or not if the value is greater than zero then it will be global frequent otherwise is infrequent.
Step7: Like that the entire site will determine the will determine the global support site3, site4 site5…….site n.
Step8: Finally the site1 determine the actual support by using the formula
$AS_j = PS_j$ + mask value
Step9: At last the site1 will send the determined value of actual support and global frequent item set to all other site in the horizontal partition.
Step10: Each site will generate the association rule by using their confidence value.
Site 1 Determine the mask value by using the following formula
Mask value is determined by using two different hash functions one after another
Key1=Hash (key) =key mod N
And after that
Mask key = Hash2 (Key1) =Key+M^{key1}

Figure 2: Ck secure sum algorithm for association rule mining

Double hash function [10][11] is used to find mask value enhance the privacy making the partial support (association rule) in more disguised form. When sites try to find the global frequent item sets from its local frequent item sets then the sites also includes some of the infrequent item sets to its local frequents items and send to the next sites so that the current site unable to know the frequent item set of previous site. This mix the upcoming site notable to know the frequent item sets for the previous sites which protect the data from third party. We have constructed the movement of sites by the above algorithm.

First round



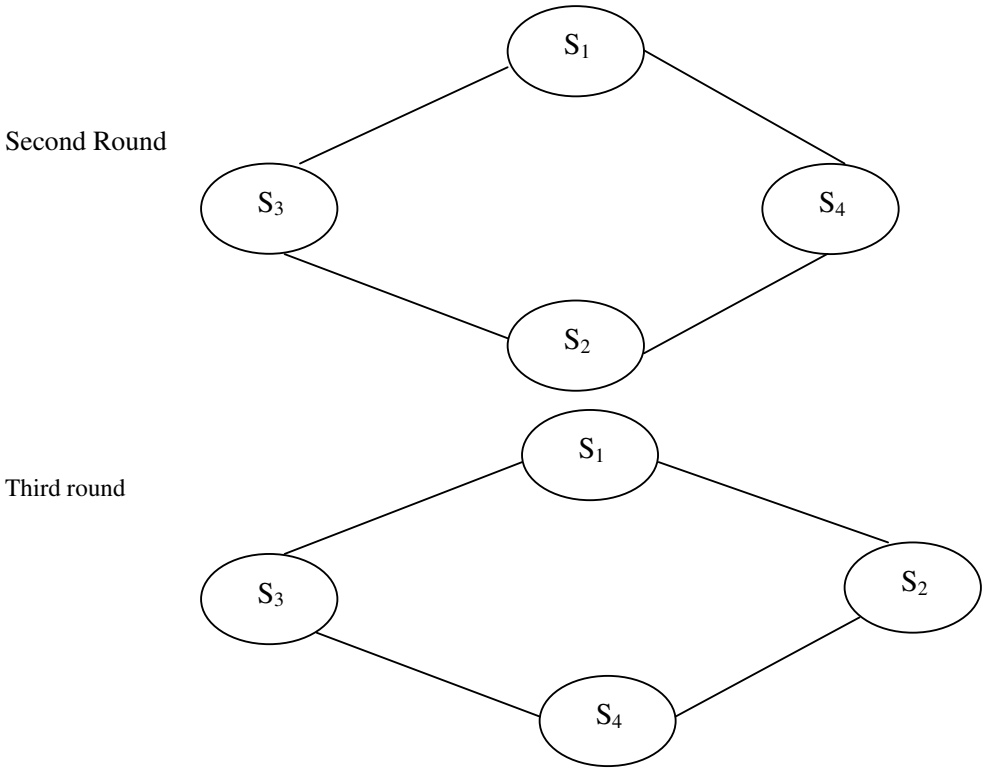


Figure 3. Represent the movement of each site in Ck secure sum protocol

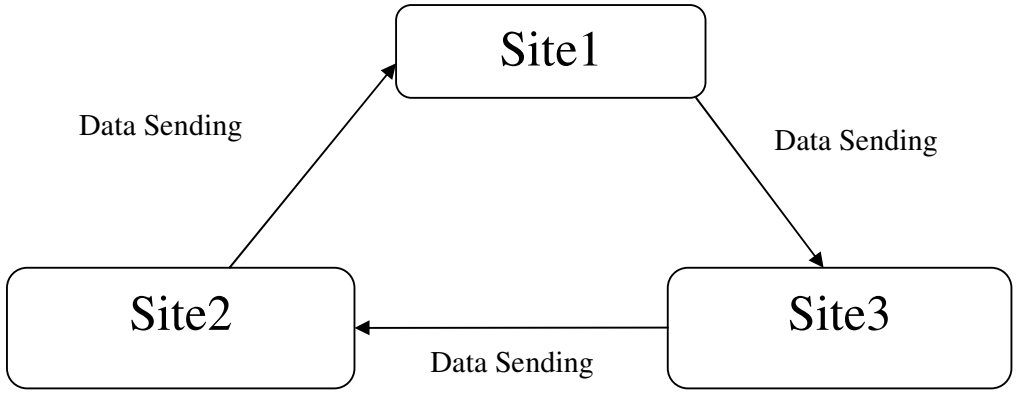


Figure 4. Communication among different sites

4. RESULTS AND EXPERIMENTATION

The proposed model is illustrated by using four horizontally partitioned distributed databases for finding privacy preserving association rule mining using Ck Secure sum protocol. In this sample model, the horizontally partitioned databases [10] [11] called fragments such as DB_1, DB_2, DB_3 and DB_4 are placed in $site_1$, $site_2$, $site_3$ and $site_4$ respectively. In this model that every time sites will change their position. Every other site maintains a database table which are partitioned horizontally given as follows.

Table 1. site1 has the data items

T-id/ item	A1	A2	A3	A4
T1	1	0	1	0
T2	0	1	0	1
T3	1	1	1	0
T4	0	1	0	1
T5	1	0	1	1

Table 2. site2 has the data items

T-id/ item	A1	A2	A3	A4
T1	1	1	1	0
T2	0	0	1	0
T3	1	1	1	1
T4	0	1	0	1
T5	1	0	1	1

Table 3. site3 has the data items

T-id/ item	A1	A2	A3	A4
T1	1	0	1	0
T2	1	0	0	1
T3	1	0	1	0
T4	0	0	0	1
T5	1	0	1	0
T6	1	0	0	1

Table 4. site4 has the data items

T-id/ item	A1	A2	A3	A4
T1	1	0	1	0
T2	1	0	1	1
T3	0	1	0	1
T4	1	1	0	1
T5	0	1	0	0

Let the minimum support is 40% for the entire database
At site 1: The list of frequent item at site 1 {A1, A2, A3, A4, (A1, A2), (A1, A3), (A2, A3), (A2, A4), (A1, A2, A3)}
At site 2: The list of frequent item at site2 {A1, A2, A3, A4, (A1, A2), (A1, A3), (A1, A4), (A2, A3), (A2, A4), (A3, A4), (A1, A2, A3), (A1, A3, A4)}
At site 3: The list of frequent item at site3 {A1, A3, A4, (A1, A3)}
At site 4: The list of frequent item at site4 {A1, A2, A3, A4, (A1, A3), (A1, A4)}

Consider the item set {A2}
Select the random number RN1=10, RN2=20, RN3=10, RN4=10
Key =110, M=2
Hash key=key mod M
Mask key=hash key- M^{key}
Hash key=110 mod 2=0
Mask key=110-2^0=109
For item set I= {A2}
PS=I1Support- Minimum support*DB + (RN I -RN (i-1)) + Mask key

Round 1-
PS11= 3-.4*5+ (10-20) +109=100
PS12=3-.4*5+ (20-10) +100=111
ps13=0-.4*6+ (10-20) +111=98.6
ps14=3-.4*5+ (10-10) +98.6=99.6
Global encrypt support (GES) = Partial support-Mask key
GES= 99.6-109=-9.4
Round 2 -
PS11= 3-.4*5+ (10-20) +109=100
ps13=0-.4*6+ (10-20) +100=87.6
PS12=3-.4*5+ (20-10) +87.6=98.6
ps14=3-.4*5+ (10-10) +98.6=99.6
Global encrypt support (GES) = Partial support-Mask key
GES= 99.6-109=-9.4

Round 3 -
PS11= 3-.4*5+ (10-20) +109=100
ps13=0-.4*6+ (10-20) +100=87.6
ps14=3-.4*5+ (10-10) +87.6=88.6
PS12=3-.4*5+ (20-10) +86.6=99.6
Global excess support (GES) = Partial support-Mask key
GES= 99.6-109=-9.4

If global excess support is greater than or equal to Zero than it is frequent otherwise it is infrequent. So in this problem the value of global excess support is negative, it means it is globally infrequent. But A2 is frequent at site1, site2 and site4 so it's locally frequent and infrequent in site3. A2 make it is a frequent after doing some of the computation in various sites. And after adding the dummy item sets that convert infrequent to frequent or frequent to infrequent that makes that the successor sites will never able to know the previous result value of other sites.

5. CONCLUSIONS

This paper addresses the problem of computing association rules within a scenario of homogeneous database. We assume that all sites have the same schema, but each site does not have information on different entities. The goal is to produce association rules that hold globally while limiting the information shared about each site. Many proposals have been sited to implement SMC. SMC being used in large scale databases which extends to preserve privacy to the private data of different sites. In this paper our focus is based on horizontal partitioned distributed data through a popular association rule mining technique.

6. REFERENCES

[1]Agrawal, R., et al "Mining association rules between sets of items in large database". *In: Proc. of ACM SIGMOD'93, D.C, ACM Press,* Washington, pp.207-216, 1993.

[2]. Agarwal, R., Imielinski, T., Swamy, A. "Mining Association Rules between Sets of Items in Large Databases*", Proceedings of the 1993 ACM SIGMOD International Conference on Management of Data,* pp. 207-210, 1993.

[3]. Srikant, R., Agrawal, R "Mining generalized association rules", *In: VLDB'95*, pp.479-488, 1994.

[4]Agrawal, R., Srikant, R, "Privacy-Preserving Data Mining", *In: proceedings of the 2000 ACM SIGMOD on management of data*, pp. 439-450, 2000.

[5] Lindell, Y., Pinkas, B, "Privacy preserving Data Mining", *In: Proceedings of 20th Annual International Cryptology Conference (CRYPTO), 2000.*

[6]Kantarcioglu, M., Clifton, C, "Privacy-Preserving distributed mining of association rules on horizontally partitioned data", *In IEEE Transactions on Knowledge and Data Engineering Journal*, *IEEE Press*, Vol 16(9), pp.1026-1037, 2004.

[7] Han, J. Kamber, M, "Data Mining Concepts and Techniques". Morgan Kaufmann, San Francisco, 2006.

[8]Sheikh, R., Kumar, B., Mishra, D, K, "A Distributed k- Secure Sum Protocol for Secure Multi-Site Computations". *Journal of Computing*, Vol 2, pp.239-243, 2010.

[9]Sugumar, Jayakumar, R., Rengarajan, C "Design a Secure Multi Site Computation System for Privacy Preserving Data Mining*". International Journal of Computer Science and Telecommunications*, Vol 3, pp.101-105. 2012.

[10] N V Muthu Lakshmi, Dr. K Sandhya Rani ,"Privacy Preserving Association Rule Mining without Trusted Site for Horizontal Partitioned database", *International Journal of Data Mining & Knowledge Management Process (IJDKP)* Vol.2, pp.17-29, 2012.

[11] N V Muthu lakshmi, Dr. K Sandhya Rani, "Privacy Preserving Association Rule Mining in Horizontally Partitioned Databases Using Cryptography Techniques*", International Journal of Computer Science and Information Technologies(IJCSIT)*, Vol. 3 (1) , PP. 3176 – 3182, 2012.

[12] Goldreich, O., Micali, S. & Wigerson, A. ,"How to play any mental game", *In: Proceedings of the 19th Annual ACM Symposium on Theory of Computing,* pp.218-229, 1987.

[13] Franklin, M., Galil, Z. & Yung, M.,"An overview of Secured Distributed Computing". *Technical Report CUCS- 00892, Department of Computer Science*, Columbia University.

[14]Goldreich,O.,"SecureMultiparty Computation", available form *http://www.wisdom.weizmann.ac.il/home/oded/publichtml/foc.html*

Authors

Jayanti Danasana
School of Computer Science, KIIT University,
Bhubaneswar, ODISHA, INDIA
jayantifcs@kiit.ac.in

Raghvendra Kumar
School of Computer Science, KIIT University,
Bhubaneswar, ODISHA, INDIA
raghvendraagrwal7@gmail.com

Debadutta Dey
School of Computer Science, KIIT University,
Bhubaneswar, ODISHA, INDIA
dev.dey009@gmail.com