Preserving Privacy of Association Rule Mining in Horizontally Partition Database

Mitali R. Jawarkar[1] Mohseen Ahmed[2]
[2]Assistant Professor
[1,2,]Department of Computer Science & Engineering
[1,2]Mgm College of Engineering, Nanded, India

Abstract— Data mining is very important and most growing area today and that is used to find or extract important knowledge from large data collection. These data collection are partitioned among various sites or parties. Privacy may avoid the parties from directly sharing the data. So, we use a protocol for preserving privacy of association rule mining in horizontally partition database. The current leading protocol is Kantarcioglu and Clifton as known a K & C protocol. This is based on unsecured distributed version of the apriori algorithm named as Fast Distributed Mining (FDM) algorithm of Cheung et al. The main part of this protocol has two novel secure multi- party algorithm one that computes the union of private subset that is each of interacting player hold and another that tests whether an element held by one player is included in a subset held by another. In that setting there are various sites or player that hold homogeneous database, means that share the same schema but hold the information on different entities. The main goal of this is to find out frequent itemsets and association rule with minimum support threshold and minimum confidence threshold. Data mining association rule technique is used for discovering or finding interesting relations in large database. Support is how frequently a specific item or itemsets are present in database and confidence is an indication of how often the rule has been found to be true. This protocol offers enhanced privacy with respect to early protocol. In addition it is not complicated and it is very vary effective in the terms of communication cost, communication round and last computational cost.

Key words: Data Mining, Rule Mining, K & C Protocol

I. INTRODUCTION

We live in the world in that world huge amount of data should be collected daily. WE study here the problem of secure mining of association rules in horizontally partitioned databases. In that setting, there are several sites (or players) that hold homogeneous databases, i.e., databases that share the same schema but hold information on different entities. The goal is to find all association rules with support at least s and confidence at least c, for some given minimal support size s and confidence level c, that hold in the unified database, while minimizing the information disclosed about the private databases held by those players. The information that we would like to protect in this context is not only individual transactions in the different databases, but also more global information such as what association rules are supported locally in each of those databases. That goal defines a problem of secure multi-party computation. In such problems, there are M players that hold private inputs, $x_1; \ldots ; x_M$, and they wish to securely compute $Y= f(x_1; \ldots ; x_M)$ for some public function f. If there existed a trusted third party, the players could surrender to him their inputs and he would perform the function evaluation and send to them the resulting output. In the absence of such a trusted third party, it is needed to devise a protocol that the players can run on their own in order to arrive at the required output y. Such a protocol is considered perfectly secure if no player can learn from his view of the protocol more than what he would have learnt in the idealized setting where the computation is carried out by a trusted third party.

Kantarcioglu and Clifton studied that problem and devised a protocol for its solution. The main part of the protocol is a sub-protocol for the secure computation of the union of private subsets that are held by the different players.

II. LITERATURE REVIEW

A. Privacy-preserving Distributed Mining of Association Rules on Horizontally Partitioned Data

Data mining can this article address the rules of safe mining data associated horizontal partition. The method includes encryption technology to minimize the sharing of information, while adding little overhead to the mining task. Data mining technology has been determined from large data patterns and trends as a means. Data mining and data warehousing to go hand in hand: most tools operate, all collected data to a central site, and then run the algorithms of these data. Extract important knowledge from large data collections – but sometimes these collections are split among various parties. Privacy concerns may prevent the parties from directly sharing the data [4], and some types of information about the data. However, privacy concerns can prevent building a centralized warehouse – data may be distributed among several custodians, none of which are allowed to transfer their data to another site. This paper addresses the problem of computing

B. Distributed Databases Rules of Mining in the Field of Association

Aim of data mining is to extract vital information from massive datasets, however typically these datasets are split among varied parties. Data mining is defined as the technique for extracting hidden, predictive and knowledge data from large distributed databases. The technology that has emerged as method of identifying patterns and trends from large quantities of knowledge. This paper studies the matter of association rule mining in horizontally distributed databases. In the distributed databases, there are many players that hold same databases that share same schema however hold data totally on different entities. The goal is to search out all association rules with support s and confidence c to attenuate the data disclosed regarding the personal databases command by those players [1].

C. Anonymization of Centralized and Distributed Social Networks by Sequential Clustering

Efficient Computation anonymizations problem Partitioned database. Given shared across multiple sites, horizontal or vertical, we have designed a secure distributed algorithms, so that different locations get k Check database - Anonymous ℓ - various data bases of their union not to divulge sensitive information. Our algorithm is based on [7] sequential algorithms, and provide anonymizations Gongyongshiye than other anonymization algorithms, especially those so far implemented in a distributed environment significantly better. Our algorithm can be applied to a wide variety of technical and practical measures and a number of sites. While previous cryptographic algorithms distributed algorithms rely on expensive, password assume that our solution is surprisingly minimal.

III. PRELIMINARIES

A. Definitions and Notations

Let D be a transaction database. We view D as a binary matrix of N rows and L columns, where each row is a transaction over some set of items A ={a_1 . . . . a_L}, and each column represents one of the items in A. The database D is partitioned horizontally between M players denoted P1 . . . . PM. Player Pm holds the partial database Dm that contains Nm =|Dm| of the transactions in D, 1 < m < M. The unified database is D = D1 U…. UDM, and it includes N= $\sum_{m=1}^{M}$ Nm transactions. An item set X is a subset of A. Its global support, supp(X),is the number, of transactions in D that contain it. Its local support suppm(X) is the number of transactions in Dm that contain it. Let s be a real number between 0 and 1 that stands for a required support threshold. An item set X is called s-frequent if supp(X)> sNm. It is called locally s-frequent at Dm , if supp(X)> sNm. For each 1 <k < L, Let F_s^k denote the set of all k-item sets that are s-frequent, and $F_s^{k,m}$ be the set of all k-item sets that are locally s-frequent at Dm, 1 <m < M. Our main computational goal is to find, for a given threshold support 0 < s <1, the set of all s-frequent item sets, $F_s = \bigcup_{k=1}^{L} F_s^k$ We may then continue to find all association rules of support at least sN and confidence at least c. Computing association rules without disclosing individual transactions is straightforward.

Let D be a transaction database. We view D as a binary matrix of N rows and L columns, where each row is a transaction over some set of items A ={a1 . . . . aL}, and each column represents one of the items in A. The database D is partitioned horizontally between M players, denoted P1 . . . . PM. Player Pm holds the partial database Dm that contains Nm =|Dm| of the transactions in D, 1 < m < M. The unified database is D = D1 U…. UDM, and it includes N= $\sum_{m=1}^{M}$ Nm transactions. An item set X is a subset of A. Its global support, supp(X), is the number, of transactions in D that contain it. Its local support suppm(X) is the number of transactions in Dm that contain it. Let s be a real number between 0 and 1 that stands for a required support threshold. An item set X is called s-frequent if supp(X)> sNm. It is called locally s-frequent at Dm , if supp(X)> sNm. For each 1 <k < L, Let F_s^k denote the set of all k-item sets that are s-frequent, and $F_s^{k,m}$ be the set of all k-item sets that are locally s-frequent at Dm, 1 <m < M. Our main computational goal is to find, for a given threshold support 0 < s <1, the set of all s-frequent item sets, $F_s = \bigcup_{k=1}^{L} F_s^k$ We may then continue to find all association rules of support at least sN and confidence at least c. Computing association rules without disclosing individual transactions is straightforward.

IV. THE FAST DISTRIBUTED MINING ALGORITHM

The main idea is that any s-frequent item set must be also locally s-frequent in at least one of the sites. Hence, in order to find all globally s-frequent item sets, each player reveals his locally s-frequent item sets and then the players check each of them to see if they are s-frequent also globally.

The FDM algorithm proceeds as follows:

Algorithm - Fast Distributed Mining (FDM)

- Initialization
- Candidate Sets Generation
- Local Pruning
- Unifying the candidate item sets
- Computing local supports
- Broadcast Mining Results

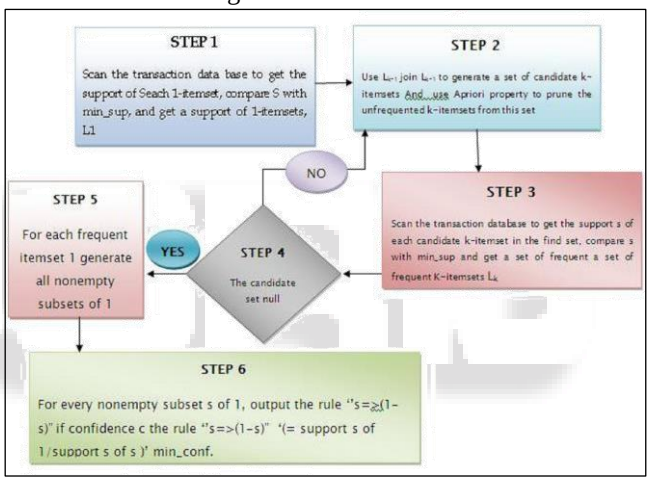

Fig. 1: Architecture of advanced FDP

Example

Let D be a database of N = 18 item sets over a set of L =5 items, A={1,2,3,4,5} It is partitioned between M =3 Players and the corresponding partial databases are:

D1={12,12345,124,1245,14,145, 235, 24,24},

D2 ={1234,134,23,234,2345},

D3 {1234,124,134,23}

For example, D1 includes N1 = 9 transactions, the third of which consists of three items—1, 2 and 4.

Setting s =1/3, an item set is s-frequent in D if it is supported by at least 6 =sN of its transactions. In this case,

F^1_s = {1, 2, 3, 4},

F^2_s = {12,14,23,24,34},

F^3_s = {124},

F^4_s = {Ø}

and Fs = $F^1_s \cup F^2_s \cup F^3_s$ For example, the item set 34 is indeed globally s-frequent since it is contained in 7 transactions of D. However, it is locally s-frequent only in D2 and D3.

In the first round of the FDM algorithm, the three players compute the sets $C^{1,m}{}_s$ of all 1-item sets that are locally frequent at their partial databases:

$C^{1,1}{}_s$={1,2,4,5}, $C^{1,2}{}_s$ ={1,2,3,4}, $C^{1,3}{}_s$ ={1,2,3,4}.

Hence, $C^1{}_s$ ={1,2,3,4,5} Consequently, all 1-item sets have to be checked for being globally frequent; that check reveals that the subset of globally s-frequent 1-item sets is $F^1{}_s$ ={1,2,3,4}.
In the second round, the candidate item sets are:
$C^{2,1}{}_s$ ={12, 14, 24},
$C^{2,2}{}_s$ ={13, 14, 23, 24, 34},
$C^{2,3}{}_s$ ={12,13, 14,23,24,34}.
Hence, $C^2{}_s$ ={12, 13, 14,23, 24, 34}. Then, after verifying global frequency, we are left with $F^2{}_s$ {12, 14, 23, 24,34}.
In the third round, the candidate item sets are:
$C^{3,1}{}_s$ ={124},
$C^{3,2}{}_s$ ={ 234},
$C^{3,3}{}_s$ ={124}. So, $C^3{}_s$ ={124,234} and then $F^3{}_s$ ={124}.

V. MODULES

1) Privacy Preserving Data Mining
2) Distributed Computation
3) Frequent Item sets
4) Association Rules

A. Modules Description

1) Privacy Preserving Data Mining:
Wherein the data owners and data miners are two different entities , and another , in which the data is distributed between several parties [11] aims to unify the corpus of data on who they are holding joint implementation data mining . In the first set, the goal is to protect the data record from the data miners. Therefore, the data of the owner intended .data prior to its release. The main approach in this context is to apply data perturbation. The idea is that. Computation and communication costs versus the number of transactions N the perturbed data can be used to infer general trends in the data, without revealing original record information. In the second setting, the goal is to perform data mining[10] while protecting the data records of each of the data owners from the other data owners. This is a problem of secure multiparty computation. The usual approach here is cryptographic rather than probabilistic.

2) Distributed Computation:
The part of the safety performance achieved in the first embodiment 13] we use to perform protocol UNIFI -KC, wherein the switching password is 1024 RSA In the second implementation (note FDM) unified step, we use our agreement UNIFI, wherein the hash function is keyed HMAC. In both embodiments, the step we have achieved in a safe manner described later FDM algorithm 5. We tested two implementations with respect to the three measures:

1) Over the entire agreement of all players (FDMKC and FDM) total computation time. The measures include apriori calculation time, and to recognize the global Sfrequent item sets, as described later in.
2) Total computation time of the unification proposed rules only (UNIFI-KC and UNIFI) over all players.
3) Total message size. We ran three experiment sets, where each set tested the dependence of the above measures on a different parameter: • N — the number of transactions in the unified database.

3) Frequent Itemsets:
They believe that two possible settings. If the desired output includes all global S- frequent item sets , as well as their support of the size , the value of _ (x) can be displayed for all .Here we describe the proposed Kantarcioglu and Clifton solutions. SNM - in this case, these values can be summed security protocol, which Pm private addend is suppm (x) is calculated. Even more interesting is set, however, is not the size of a part of the support in the desired output. We continue to discuss

4) Association Rules:
The main ingredient in the proposed rules we proposed in a new security rules for multi-party proposed to calculate private subset union (or intersection) , each player holds interactive .Once set Fs all S- frequent item sets are found , we will continue to look for all the (S, C) –association rule (rule , at least SN support and confidence , at least C). In order to derive all from fs (S, C) , we rely on simple lemma effective way -association rules.

VI. CONCLUSION

The protocol preserving privacy of association rules mining in horizontally partitioned databases offers enhanced privacy and security than other protocol [2]. There are various methods to secure private data of users from any third party which results into some output. The system will try to obtain the best results in terms of time efficiency and computation cost. Privacy preserving can be applied in different domains. The focus in this thesis is on the association rule mining domain. The goal of association rule mining is to find all patterns based on some hard thresholds, such as the minimum support and the minimum confidence. The owners of these databases might need to hide some patterns that are of a sensitive nature. The sensitivity and the degree of sensitivity are decided by data owners. We introduced an effective secure algorithm that gives the data owners the control to decide in which depth each sensitive pattern can be hidden based on the sensitivity of that patterns. We also studied the existing algorithms and stated their drawbacks.

Then, we see secure multi-party computation algorithms, specifically, those related to data mining. This field is called, privacy-preserving data mining in secure multi-party computation. We use a protocol that allows three or more parties compute their private goals.

The main goal of this is to mine frequent itemsets and association rule with minimum support and minimum confidence. This goal is fulfilled successfully.

VII. FUTURE SCOPE

The direction to future work is to devise an efficient protocol for inequality verifications that uses the existence of semi-honest third party and another in implementation of the techniques to the problem of partitioned association rule mining in vertical setting.

REFERENCES

[1] Tamir Tassa “Secure Mining of Association Rules in Horizontally Distributed Databases,” IEEE

TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 4, APRIL 2014

[2] R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules in Large Databases," Proc 20th Int'l Conf. Very Large Data Bases (VLDB), pp. 487-499, 1994.

[3] R. Agrawal and R. Srikant, "Privacy-Preserving Data Mining," Proc. ACM SIGMOD Conf., pp. 439-450, 2000.

[4] A. Ben-David, N. Nisan, and B. Pinkas, "FairplayMP - A System for Secure Multi-Party Computation," Proc. 15th ACM Conf. Computer and Comm. Security (CCS), pp. 257-266, 2008.

[5] D.W.L. Cheung, J. Han, V.T.Y. Ng, A.W.C. Fu, and Y. Fu, "A Fast Distributed Algorithm for Mining Association Rules," Proc. Fourth Int'l Conf. Parallel and Distributed Information Systems (PDIS), pp. 31-42, 1996.

[6] D.W.L Cheung, V.T.Y. Ng, A.W.C. Fu, and Y. Fu, "Efficient Mining of Association Rules in Distributed Databases," IEEE Trans. Knowledge and Data Eng., vol. 8, no. 6, Dec. 1996.

[7] A.V. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke, "Privacy Preserving Mining of Association Rules," Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 217-228, 2002.

[8] M.J. Freedman, K. Nissim, and B. Pinkas, "Efficient Private Matching and Set Intersection," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 1-19, 2004.

[9] H. Grosskreutz, B. Lemmen, and S. R€uping, "Secure Distributed Subgroup Discovery in Horizontally Partitioned Data," Trans. Data Privacy, vol. 4, no. 3, pp. 147-165, 2011.

[10] M. Kantarcioglu and C. Clifton, "Privacy-Preserving Distributed Mining of Association Rules on Horizontally Partitioned Data," IEEE Trans. Knowledge and Data Eng., vol. 16, no. 9, pp. 1026-1037, Sept. 2004.

[11] M. Kantarcioglu, R. Nix, and J. Vaidya, "An Efficient Approximate Protocol for Privacy-Preserving Association Rule Mining," Proc. 13th Pacific-Asia Conf. Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 515-524, 2009.

[12] L. Kissner and D.X. Song, "Privacy-Preserving Set Operations," Proc. 25th Ann. Int'l Cryptology Conf. (CRYPTO), pp. 241-257, 2005.

[13] Y. Lindell and B. Pinkas, "Privacy Preserving Data Mining," Proc. Crypto, pp. 36-54, 2000.

[14] A. Schuster, R. Wolff, and B. Gilburd, "Privacy-Preserving Association Rule Mining in Large-Scale Distributed Systems," Proc. IEEE Int'l Symp. Cluster Computing and the Grid (CCGRID), pp. 411-418, 2004.

[15] R. Srikant and R. Agrawal, "Mining Generalized Association Rules," Proc. Int'l Conf. Very Large Data Bases (VLDB), pp. 407-419,1995.

[16] T. Tassa and E. Gudes, "Secure Distributed Computation of Anonymized Views of Shared Databases," Trans. Database Systems, vol. 37, article 11, 2012.

[17] J. Vaidya and C. Clifton, "Privacy Preserving Association Rule Mining in Vertically Partitioned Data," Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), pp. 639- 644, 2002.

[18] A.C. Yao, "Protocols for Secure Computation," Proc. 23rd Ann. Symp. Foundations of Computer Science (FOCS), pp. 160-164, 1982.

[19] J. Zhan, S. Matwin, and L. Chang, "Privacy Preserving Collaborative Association Rule Mining," Proc. 19th Ann. IFIP WG 11.3 Working Conf. Data and Applications Security, pp. 153-165, 2005.

