Secure Extraction of Association Rule from Distributed Database

Mahale Mohini V
Department of Computer Engineering,
SNDCOE and RC Yeola, Nashik-423401

Shaikh I.R.
Department of Computer Engineering,
SNDCOE and RC Yeola, Nashik-423401

ABSTRACT
There are many techniques to extract association rules from large datasets, but sometimes these datasets are distributed horizontally which is called strew database. In the strew database there are several sites or players that hold homogeneous database this database shares the same schema but hold information on different entities. For extracting association rules from such database the existing system is not so secure and efficient. The proposed system given here provides a secure and efficient solution for the problem stated above. Here we are going to use Fast Distributed mining (FDM) which is an unsecured distributed version of the Apriori algorithm. The proposed system gives enhanced version of FDM. Which offers enhanced privacy with respect to the protocol in [1] Also, it is more simple and significantly more effective in terms of communication rounds, communication cost and computational cost.

Index Term-
Privacy preserving data mining, distributed computation, frequent item sets, association rules

1. INTRODUCTION
In strew database where datasets are horizontally partitioned, there are several players that hold identical databases, this databases share the same schema but hold information on different entities. In such scenario there is problem to find all association rules with support and confidence for some given minimal support size and confidence level that hold in the unified database. But while doing this the information of private database should not disclosed to the participating players.

In this problem there are *M* players that hold private inputs, y1, . . . , *yM*, and they wish to securely compute z = *f*(y1, . . . , *yM*) for some public function *f*. If there existed a trusted third party, the players could give him their inputs (private datasets) 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 develop a protocol that the players can run on their own in order to reach at the required output z. Such a protocol is considered perfectly secure so that another player cannot learn the extra information in the absence of trusted third party. Kantarcioglu and Clifton gives solution to this problem [1]. 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. This is also the only part in the protocol in which the players may extract from their view of the protocol information on other databases, beyond what is implied by the final output and their own input.

We propose here an alternative protocol for the secure computation of the union of private subsets. This protocol is more efficient than in [1] in terms of simplicity and efficiency as well as privacy. Our protocol does not depend on oblivious transfer and commutative encryption. This solution is still not perfectly secure, it leaks excess information only to a small number of possible coalitions, unlike the protocol of [1] that discloses information also to some single players. This protocol may leak information that is less important than of protocol [1].

2. LITERATURE SURVEY
There are many Privacy Preserving Association Rule Mining (PPARM) algorithms are proposed for different partitioning methods by satisfying privacy constraints. The various methods such as randomization, perturbation, heuristic and cryptography techniques are proposed by different authors to find privacy preserving association rule mining in strew databases and vertically partitioned databases. In the case of secure multiparty computation while computing the association rules, the data of participating parties should nod disclose to each other .There are many solutions to satisfy the above constrained. The first solution was propose by Yao [2] this technique was only suitable for two players.

Latter in paper [2] specifies the protocol for secure mining of association rules in horizontally partioned databse, where Fast Distributed Mining algorithm (FDM) is get uesd for mining of association rules. In this protocol players finds their locally locally frequent itemsets then the players check each of them to find out globally s-frequent item set. But the protocol assumes that the players are semi honest; they try to extract information. Hence the player calculate encryption of private database collectively by applying commutative encryption. This protocol offers better privacy and is significantly more efficient in terms of communication cost and computational cost than the pravious one .But this solution is not perfectly secure cause it leaks excess information.

In the problem of extracting association rules from strew database the goal is to perform data mining while protecting the data records of each of the data owners and from the other data owners. computation. The typical approach here is cryptographic instead of probabilistic. Lindell and Pinkas [3] gives the solution by implementing secure ID3 decision tree. Secure clustering using the EM algorithm was implemented by Lin et al. [4] over horizontally distributed data. In The problem of distributed association rule mining was studied in [5], and [6] but here the data was distributed vertically, where each party holds a different set of attributes. Also the work of [7] considered this problem in the horizontal setting, but they considered large-scale systems in which, on top of the parties that hold the data resources there are also managers which are computers that assist the resources to decrypt messages. There is another solution given in[20] this protocols uses the homomorphic encryption, while our protocol uses commutative encryption, the computational costs by using homomorphic encryption are significantly higher than using the commutative encryption.

The paper [8] also provides survey of association rule based techniques for privacy preserving where it studied on three methods i.e. heuristic-based technique, Cryptography based techniques and Reconstruction-based techniques. A heuristic-based technique depends on adaptive modification which modifies only selected values that minimize the utility loss with the help of Centralized Data Perturbation-

3. EXISTING SYSTEM

The proposed system gives the alternative protocol which will overcome from the problem which are occurred in Farst distributed Mining (FDM) proposed by Kantarcioglu and Clifton(1).The proposed system is more efficient than the existing system in terms of privacy, communication rounds, communication cost and computational cost. The existing and proposed system both are based on FDM [1], which is an unsecured version of the Apriori algorithm. The proposed system computes a parameterized family of function which is called as threshold function In which two cases correspond to the problems of computing the union and intersection of private subsets. The protocol used for this function can be used in other cases as well. The major problem of extraction of association rule is set inclusion problem; the problem where Bob holds a private subset of some ground set, and Alice holds an element in the ground set, and they wish to determine whether Alice's element is within Bob's subset, without revealing to either of them information about the other party's input beyond the above described inclusion.

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

The FDM algorithm proceeds as follows:

1) Initialization: All the players should calculate all k-item sets that are s-frequent that is calculate F^k_s.

2) Generation of candidate set: The set of all local and global frequent item sets are get calculated by each player P_m. Specifically Pm computes $F^{k-1,m}_s \cap F^{k-1}_s$. Then the Apriori algorithm is get performed to generate the set $B^{k,m}_s$.

3) Local Pruning: Each player computes $supp_m(X)$. He then maintains only locally frequent item which is denoted by $Cs^{k,m}$.

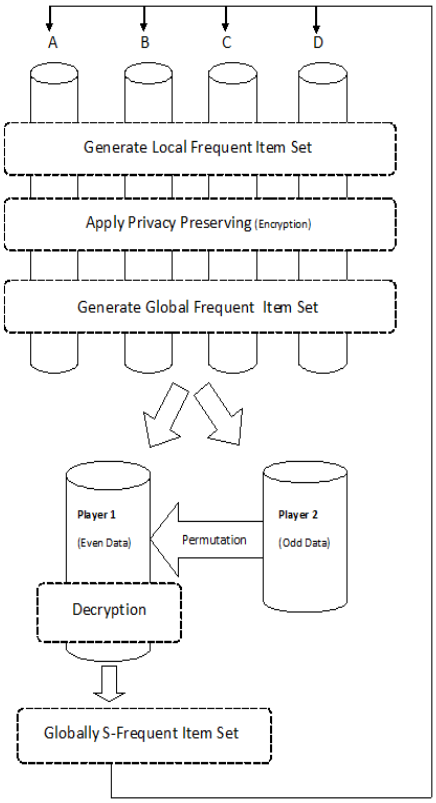
4) Unifying the candidate item sets: Each player broadcasts his own set of items $Cs^{k,m}$ which is calculated in above step. Then all players computes $C^k s$

5) Computing local supports: Local supports of all item sets that is $C^k s$ is get calculated.

6) Broadcast mining results: Each player broadcasts his own local support. So that everyone can compute the global support of every item set. Finally the set of all globally frequent item sets F^k_s whish is subset of C^k_s is get produced.

4. PROPOSED SYSTEM

The FDM algorithm violates privacy in two stages: In Step 4, where the players broadcast the item sets that are locally frequent in their private databases, and in Step 6, where they broadcast the sizes of the local supports of candidate item sets. Our improvement is with regard to the secure implementation of Step 4, which is the more costly stage of the protocol, The alternative implementation Is better in terms of privacy and efficiency compare to previous one.



4.1 Secure Computation of Threshold Function

Each player P_m has an binary vector b_m that characterizes the private subset held by player , Then the union of the private subsets is described by the OR of those private vectors, $b=V^M_m$ bm. Such a vector can be calculated by the function which is called as threshold function. Following algorithm is used for secure function of threshold function

Algorithm for Threshold function:

1. Each P_m selects M random share vectors $b_{m,l}$,such that $\sum^M_{l=1} b_{m,l} = b \bmod (M+1)$
2. Each player P_m sends $b_{m,l}$ to P_l
3. Each P_l computes $s_l = (s_l(1)\ldots\ldots\ldots, s_l(n)) = \sum^M_{m=1} b_{m,l} \bmod (M+1)$.
4. Players Pl,$2 \le l \le M-1$, sends S_l to P_1 .
5. P_1 computes $s=(s_l(1)\ldots\ldots\ldots, s_l(n)) \sum^{M-1}_{l=1} s_l \bmod (M+1)$.
6. For i=1,……n do
7. If $(s(i)+s_M(i) \bmod (M+1) < t$ set b(i) = 0

   Otherwise set b(i)= 1.
8. End for
9. Output b= (b(1),…..,b(n)).

4.2 An Improved Protocol for the Secure Computation of All Locally Frequent Item Sets

As before, we denote by F^{k-1}_s the set of all globally frequent (k-1)item sets, and by $Ap(F^{k-1}_s)$ the set of k-item sets that the Apriori algorithm generates when applied on F^{k-1}_s. All players can compute the set $Ap(F^{k-1}_s)$ and decide on an ordering of it. (Since all item sets are subsets of $A = \{a_1,\ldots a_L\}$, they may be viewed as binary vectors in {0,1} and, as such, they may be ordered lexicographically.) Then, since the sets of locally frequent k-item sets, $C^{k,m}{}_S$, $1 \le m \le M$ are subsets of $Ap(F^{k-1}_s)$ they may be encoded as binary vectors of length $n_k := Ap(F^{k-1}_s)$

Hence, the players can compute the union by invoking threshold function their binary input vectors.

Algorithm:

Input: Each player Pm has an input subset $C^{k,m}_s$

Output: C^k_s = Union $C^{k,m}_s$

1. Each player Pm encodes his subset $C^{k,m}_s$

   As a binary vector b_m of length $nk = |Ap(F^{k-1}_s)|$,in accord with the agreed ordering of $Ap(F^{k-1}_s)$.

2. The players invoke Protocol threshold to compute $b=T_1(b_1,\ldots\ldots\ldots,b_M) = OR\ b_m$

3. C^k_s is the subset of $Ap(F^{k-1}_s)$ that is described by b.

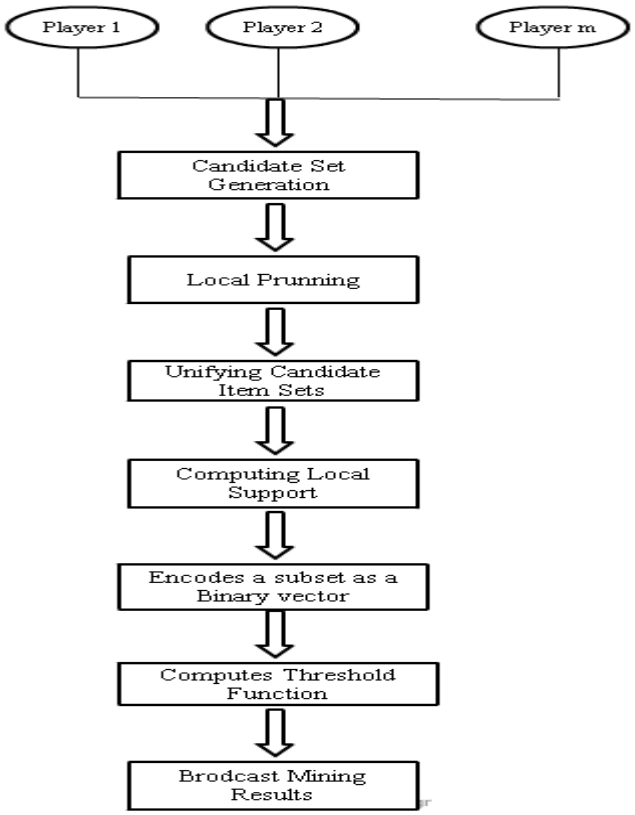


Fig2. Architecture of Proposed System

5. CONCLUSION

Extracting association rules from strew database involves the problem of secure multiparty communication. We proposed a protocol for secure mining of association. Rules from strew database that improves expressively upon the current leading protocol in terms of privacy and efficiency. The main ingredient is a protocol that tests the inclusion of an element held by one player in a subset held by another. n addition, this system Also, it is more simple and significantly more effective in terms of communication rounds, communication cost and computational cost.

6. REFERENCES

[1] 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.

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

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

[4] X. Lin, C. Clifton, and M.Y. Zhu, "Privacy-Preserving Clustering with Distributed EM Mixture Modeling," Knowledge and Information Systems, vol. 8, pp. 68-81, 2005.

[5] 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.

[6] 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.

[7] 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,y 2004.

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

[9] T. ElGamal, "A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms," IEEE Trans. Information Theory, vol. IT-31, no. 4, July 1985.