Scientific Journal of Impact Factor (SJIF): 4.14
e-ISSN (O): 2348-4470
p-ISSN (P): 2348-6406

International Journal of Advance Engineering and Research Development

Volume 3, Issue 2, February -2016

Survey Paper on Secure Data Mining for Horizontally Distributed DataBase

Ms. Shweta Gandevia [1], Mr. Mohit Patel[2]

[1]*Department of Computer Science and Engineering, PIT Vadodara,*
[2]*Department of Computer Science and Engineering , PIT Vadodara,*

Abstract - Data mining is used to extract important knowledge from large datasets, but sometimes these datasets are split among various parties.Privacy liability may prevent the parties from directly sharing the data and some types of information about the data.This paper presents different methods for secure mining of association rules in horizontally distributed databases. The main aim of this paper is protocol for secure mining of association rules in horizontally distributed databases. The current main protocol is that of Kantarcioglu and Clifton. This protocol, like theirs, is based on the Fast Distributed Mining (FDM) algorithm of Cheung et al., which is an unsecured distributed version of the Apriori algorithm. The main components in this protocol are two novel secure multi-party algorithms — one that computes the union of private subsets that each of the interacting players hold, and another that tests the inclusion of an element held by one player in a subset held by another. This protocol offers improved privacy with respect to the protocol in. In addition, it is simpler and is significantly more efficient in terms of communication rounds, communication cost and computational cost.

Keywords - *Privacy preserving data mining,Distributed computation,Frequent item sets,Multi-party.*

I. INTRODUCTION

Data mining is defined as the method for extracting hidden predictive information from large distributed databases. It is new technology which has emerged as a means of identifying patterns and trends from large quantities of data.The final product of this process being the knowledge, meaning the significant information provided by the unknown elements [2].

In the distributed databases, there are several players that hold homogeneous databases which share the same schema but hold information on different entities. The goal is to find all association rule with support s and confidence c to minimize the information disclosed about the private databases held by those players [1].

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.

In previous year various techniques are applied for secure mining of association rules in horizontally partitioned database. These approaches use various techniques such as data perturbation, homo-morphic encryption, keyword search and oblivious pseudorandom functions etc. These privacy preserving approaches are inefficient due to

- Homo-morphic encryption
- Higher computational cost
- In some of the techniques data owner tries to hide data from data miner.

Kantarcioglu and clifton studied the problem where more suitable security definitions that allow parties to choose their desired level of security are needed, to allow effective solutions that maintain the desired security [2]. So they devised a protocol for its solution.The main part of that protocol is sub protocol for secure computation of the union of private subsets that are held by the different players. It makes the protocol costly and its implementation depends upon encryption primitive's methods, oblivious transfer and hash function also the leakage of information renders the protocol not perfectly secure [1].

II. ARCHITECTURE OF DISTRIBUTED DATABASE

A Distributed Database is a database in which portions of the database are stored on multiple computers within a network. A Distributed Database is a database distributed between several sites. The reasons for the data distribution may include the inherent distributed nature of the data or performance reasons[1]. In a distributed database the data at each site is not necessarily an independent entity, but can be rather related to the data stored on the other sites[3]. Users have

access to the portion of the database at their location so that they can access the data relevant to their tasks without interfering with the work of others.

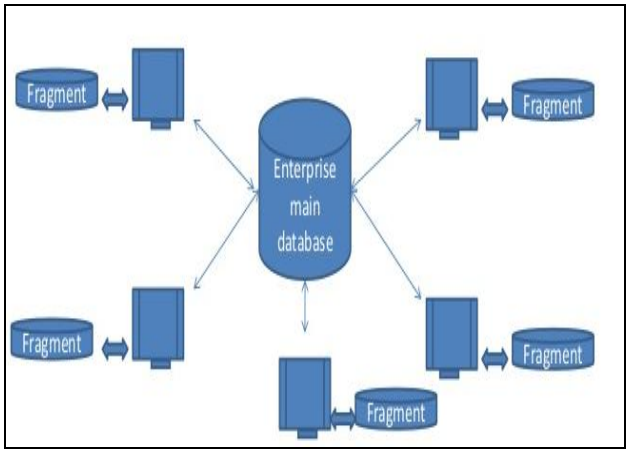

Figure 1. Basic Architecture of Distributed Database[3]

There are two types of Distributed database one is Homogeneous Distributed Database and another is Heterogeneous Distrubuted Database.

A homogeneous distributed database has identical software and hardware running all databases instances, and may appear through a single interface as if it were a single database. A heterogeneous distributed database may have different hardware, operating systems, database management systems, and even data models for different databases.

III. BASIC APPROACH

A. Definitions and Notations

Let D be a transaction database. As in [2], we view D as a binary matrix of N rows and L columns, where each row is a transaction over some set of items, $A = \{a_1, \ldots, a_l\}$, and each column represents one of the items in A. (In other words, the $\{i, j\}$ entry of D equals 1 if the i_{th} transaction includes the item a_j, and 0 otherwise.) The database D is partitioned horizontally between M players, denoted $P_1, \ldots, P_M$. Player P_m holds the partial database D_m that contains $N_m = |D_m|$ of the transactions in D, $1 \le m \le M$. The unified database is $D = D_1 \cup \ldots D_M$, and it includes $N := \Sigma^M_{m=1} N_m$ 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, $supp_m(X)$, is the number of transactions in D_m that contain it. Clearly, $supp(X) = \Sigma^M_{m=1} supp_m(X)$. 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 $supp(X) \ge {}_sN$. It is called locally s-frequent at D_m if $supp_m(X) \ge {}_sN_m$.

B. The Fast Distributed Mining Algorithm

The protocol of [2], as well as ours, are based on the Fast Distributed Mining(FDM) algorithm of Cheung et al.[4],which is an unsecured distributed version of the Apriori algorithm. Its 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:

Step 1 : Initialization

Step 2 : Candidate Sets Generation

Step 3 : Local Pruning

Step 4 : Unifying the candidate item sets

Step 5 : Computing local supports

Step 6 : Broad cast mining results

IV. SECURE MINING PROTOCOLS

Segmentation methods can be classified into four kinds, and the selection of proper method depends on the specific application and working environments. These methods are pixel, edge, region, and model based segmentation.

A. UniFI-KC Protocol

UniFi-KC Protocol[2] is proposed by Kantarcioglu and Clifton for calculating unified list of all locally frequent item sets and they wish to continue and calculate from which is set of all (k-1) globally s-frequent item sets.UniFI-KC Protocol is used for,to secure the locally frequent item sets using private key and hash function and remove the faked item sets X_m which are generated at time encryption.

Steps of UniFI-KC protocol is as proceed:

1. Select the cryptographic primal
 - Player chooses commutative cipher and its consistent random secret keyKm to each player Pm.
 - For encryption, players choose a hash function h to Apply on all item sets.
 - The players construct a lookup table with hash values of all candidate item sets which is from $Ap(F^{k-1}_s)$.

2. Encrypts the all candidate item sets
 - Players hashed all item sets in $^{Ck,m}_s$ and encrypts that Using a secret random key Km.
 - Players adds to consequent set Xm which is faked item Sets till its size becomes $| Ap(F^{k-1}_s)|$
 - Player transmits the arrangement of Xm to next Player And takes the arrangement of Xm-1 from previous player for M-1
 times.
 - Player calculates a newconsequent set X_m by encoding the previous player's consequent set $_{Xm-1}$ using random secret key Km.
 - Player hold an encryption of hashed candidate set$C^{k,m+1}_s$ using all M players.

3. Combining the Item sets
 - Player combines the list of encrypted item sets and computes the union of private subset.
 - For combining item sets, firstly combine the each odd And even players and that are sends his encrypted set to player P1 and
 P2 respectively.
 - P1 combines the item sets list which is sent by odd and even players and removes duplicates from that list. The final list
 denoted by EC^k_s.

4. Decrypts the candidate item sets
 - Player decrypts all item sets in EC^k_s, using secret random key Km, the consequent set by C^k_s.
 - For replacing hashed values with actual item sets an identifying and removing the fake item sets, player operates the lookup
 table T. Then retrieves C^k_s.
 - Pm transmits C^k_s to all his peers.

B. t-Threshold protocol

Protocol Threshold is a secure multi party protocol for computing the OR of private binary vectors. The UniFI-KC protocol safely calculates the union of private subsets of publicly known ground set $Ap(F^{k-1}_s)$.That problem is similar to problem of calculating OR of private vectors. Actually, if the ground set is, $\Omega=\{\omega 1,\ldots,\omega n\}$then any subset B of Ω may be described band employs less cryptographic primitives. The Protocol t-Threshold computes a larger range of functions, is known as threshold functions.

Steps of Threshold protocol are as follows:

1. Firstly player chooses the random shares in input binary vector and sends the consequent share to all other players.
2. Each player calculates the sl by adding the shares and sends to P1.
3. P1calculates s by adding the all slof M-1 players.
4. Players P1 and PM hold additive shares of the sum vector a:P1 has s, PM has S_M.
5. The set b(i)=0, $1\leq i \leq n$, if $s(i)+sM(i) \bmod (M+1)<t$ otherwise set b (i) =1.

C. SetInc Protocol

Protocol SetInc included three players: P1 has a vectors of elements in some ground set Ω, P_M has a vector Θ of subsets of that ground set and the output which is required as a vector B that describes the consequent set inclusions in the following manner:b(i)=0 if s(i)∈Θ(i) and otherwise b(i)=1, where $1 \leq i \leq n$.The calculation in the protocol includes a third player P2.
Steps for SetInc Protocol as proceed:

1. Players P1 and PM agree on a keyed hash function, and a corresponding secret key K.
2. Player P1 converts his sequence of elements into corresponding signatures,s', where s' is the keyed hash function of s and P_M
   does a similar conversions to the subsets which that he holds.
3. Player P1 sends s' to P2, and PM sends Θ' to P2 the Subsets Θ(i), $1 \leq i \leq n$, the elements within each subset are randomly
   arranged.
4. Player P2 performs the significant inclusion verifications on the signature values. If he discover that for a given $1 \leq i \leq n$,P2
   sets,b(i)=0, if , s'(i) ∈Θ'(i) otherwiseb(i)=1.
5. Player P2transmits vector b.

The liability of Protocol THRESHOLD-C to association is not important because of two reasons:

i. The sum vector entries do not show information about particular input vectors.
ii. The players P1, PM and P2areconspire together to study information else where the intention of the protocol.

D. An Improved Protocol(UniFI Protocol)

An improved protocol is used for the secure calculation of all locally frequent item sets. The set of all globally frequent (k - 1)-item sets denoted by$Ap(F^{k-1}_s)$.Apriori algorithm applied on F^{k-1}_s and generates the set of k-item sets.The sets of locally frequent k-item sets,$C^{k,m}_s$, $1 \leq m \leq$ Mare subsets of $Ap(F^{k-1}_s)$.They may be encoded as binary vectors of length$|Ap(F^{k-1}_s)|$.The binary vector which is encoded in the union of $C^{k,m}_s$ which is the OR of the vectors that encoded form of the locally frequent item sets $C^{k,m}_s$. By invoking Threshold-C Protocol on binary input vectors, the players can calculate the union.

Steps of UniFI protocol:Securely unifying lists of all locally frequent item sets:

1. Encodethe subset $C^{k,m}_s$ by each player Pm as binary vector bm of length$|Ap(F^{k-1}_s)|$.
2. The players invoke protocol Threshold-C for calculating b which is same as OR of b_m.
3. $Ap(F^{k-1}_s)$ is the superset of C^k_s which is asserted by b.

E. Fully Secure Protocol

In the step 2-4 in FDM algorithm,the players distribute the local pruning and union calculation and, test all candidate item sets in $Ap(F^{k-1}_s)$ are globally s-frequent. That protocol is fully secure, as it exposes only the set of globally s-frequent item sets but no further information about the partial databases. As discussed in [2], such a protocol would be much more costly since it requires each player to compute the local support of $|Ap(F^{k-1}_s)|$ item sets in the kth round instead of only $|C^k_s|$ item sets which is union set of $C^{k,m}_s$. The players will execute the secure comparison protocol to verify inequality for $|Ap(F^{k-1}_s)|$ rather than only $|C^k_s|$, item sets. Both types of added operations are very costly: the time to calculate the support size relies linearly on the size of the database, while the secure comparison protocol entails a costly oblivious transfer sub protocol. $|Ap(F^{k-1}_s)|$ is much larger than $|C^k_s|$, the added calculating time in such a protocol is expected to dominate the cost of the secure computation of the union of all locally s frequent item sets as shown in [6]. Therefore, the enhanced security offered by such a protocol is accompanied by increased implementation costs.

V. CONCLUSION

Data mining describes applications that look for hidden knowledge or patterns in large amounts of data. In this we present various techniques for secure mining of association rules in horizontally partitioned distributed databases. The improved protocol get better the current leading protocol in terms of privacy. The one ingredients of two novel secure multi-party protocol is for calculating the union of private subsets that each of the interacting players hold. And another ingredient is a protocol that tests an element held by one player included in a subset held by another. That data mining

has a very important role in our life, so we use and handle it regularly. Therefore privacy and security should be provided to database.

References

[1] Tamir Tassa,"Secure mining of association rule in horizontally distributed databases" ,IEEE trans. Konwledege and Data Engg. ,Vol. 26, no.2, April 2014.

[2] 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.

[3] International Journal Of Computer Science And Applications. Vol. 6, No.2, Apr 2013 ISSN: 0974-1011," Distributed Database: A Survey".

[4] 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.

[5] J.C. Benaloh, "Secret Sharing Homomorphisms: Keeping Shares of a Secret Secret," Proc. Advances in Cryptology (Crypto), pp. 251-260, 1986.

[6] 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.

[7] "Information Security in Big Data:Privacy and Data Mining",LEI XU,CHUNXIANG,JIAN WANG,IEEE 2014.

[8] 2015 International Conference on Circuit, Power and Computing Technologies [ICCPCT]." Privacy in Horizontally Distributed Databases Based on Association Rules ". 978-1-4799-7075-9/15/$31.00©2015 IEEE.

[9] 2015 12th International Conference on Information Technology - New Generations. "Dataless Data Mining: Association Rules-based Distributed Privacy-preserving Data Mining" .978-1-4799-8828-0/15 $31.00 © 2015 IEEE.

[10] 2013 International Conference on Cloud Computing and Big Data." Privacy-Preserving Two-Party Distributed Association Rules Mining on Horizontally Partitioned Data". 978-1-4799-2829-3/13 $26.00 © 2013 IEEE.

[11] Xuan Canh Nguyen, Hoai Bac Le, Tung Anh Cao, "An Enhanced Scheme For Privacy Preserving Association Rules Mining On Horizontally Distributed Databases," In 2012 IEEE.

