ijiarec.com

ISSN: 2348-2079

FEB-2015

International Journal of Intellectual Advancements and Research in Engineering Computations

A NOVEL SECURE MINING OF ASSOCIATION RULES WITH SUBGROUP DISCOVERY IN VERTICALLY DISTRIBUTED DATABASES

Kokilavani. K, Nandhini. V, Mohan Arvainth. V, Dr. T. Kalaikumaran, M.E., Ph.D.

ABSTRACT

We propose a protocol for secure mining of association rules in vertically distributed databases. The current protocol is that of Kantarcioglu and Clifton two protocols to be used. In Our protocols, are based on the Fast Distributed Mining (FDM) algorithm of Cheung et al. In order to improve the performance of the system, proposed the subgroup detection concept in this system. In this setting, all sites use the same set of attributes and the quality of every subgroup depends on all databases. The approach finds patterns in the union of the databases, without disclosing the local databases. To the best of our knowledge, this is the first secure approach that tackles any of the supervised descriptive rule discovery tasks. In proposed scenario we will do the vertical distributed database which is nothing but different columns of a table distributed to different sites.

Keywords*:* privacy preserving data mining, vertically distributed database, association rule mining, distributed computations.

INTRODUCTION

We study here the problem of secure mining of association rules in vertically partitioned databases. In that setting, there are several players that hold homogeneous databases. A huge challenge to investigators is, how efficiently, one can mine information (knowledge) in an application from a large database using appropriate data mining techniques. A well-known study data mining method is association rule mining whose results are being successfully used in various real life applications to obtain planned decisions. Database is vertically partitioned into two or more fragments. This method is used for protect the dataset and provide more securely and privacy for the individual data owner's data sets.

Data mining can extract important knowledge from large datacollections but sometimes these collections are split among various parties. Privacy liabilitymay be preventing the parties from directly sharing the data and types of information about the data. Data mining technology has become prominent as a means of identifying patterns and trends from large quantities of data.

Data mining and data warehousing conjointly: most popular tools operate by gathering all data into a central site then running an algorithm contrary to that data. However, privacy responsibility 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. In vertically partitioned database there are several players that hold homogeneous database. The goal is to find all association rules with support at least s and confidence at the least c, for some given minimal number of support size s, and self confidence level c, that hold in the unified database, while decreasing the information disclosed about the private databases held by those players.

That goal defines a problem of secure multiparty computation. If there existed a confidential third party, the players could surrender to him their inputs and he would perform the function evaluation and send to them the subsequent output. In the time off of such a

Author for Correspondence:
Department of computer science and engineering, SNS College of Technology, Coimbatore, India
Email: kokila.0515@gmail.com, nandhinicse255@gmail.com

confidential 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 painstaking flawlessly 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 vertically partitioned database.

These approaches use various techniques such as data agitation, homomorphism encryption, keyword search and oblivious pseudorandom functions. These privacy preserving methodologies are inefficient due to, Homomorphism encryption, higher computational cost. In some of the techniques data owner tries to hide data from data miner.

Our proposed protocol based on two novel secure multiparty algorithms using these algorithms the protocol provides enhanced privacy, security and efficiency as it uses commutative encryption. In this project we propose a protocol for secure mining of association rules in vertically distributed database. This protocol is based on: FDM Algorithm which is an unsecured distributed version of the Apriori algorithm. In our protocol two secure multiparty algorithms are involved:

1. Computes the union of private subsets that each interacting players hold.

2. Tests the inclusion of an element held by one player in subset held by another.

In vertically partitioned database there are several players that hold homogeneous database. Our protocol offers enhanced privacy with respect to the current leading K and C protocol minimalism, more efficient in positions of communication rounds, and communication cost, computational cost. In our problem, the inputs are the partial databases and the required output is the list of association rules that hold in the unified database with support and confidence no smaller than the given threshold value s and c, are respectively.

METHODOLOGY

A. SUPPORT

The rule X ⇒ Y holds with support s if s% of transactions in D contain X ∪ Y. Rules that have a s greater than a user-specified support is said to have minimum support or threshold support. The support of the rule is definite as,

Sup(X) = no of transactions that contain X / total no of Transactions.

B. CONFIDENCE

The rule X ⇒ Y holds with confidence c if c% of the transactions in D that contain X also contain Y. Rules that have a c greater than a user-specified confidence is said to have minimum confidence or threshold Confidence. The confidence of a rule is definite as,

Conf(X =>Y) = sup(X U Y)/ sup(X).

SECURE COMPUTATION OF ALL LOCALLY FREQUENT ITEM SETS

A. UNIFI-KC (UNIFYING LISTS OF LOCALLY FREQUENT ITEM SETS-KANTARCIOGLU AND CLIFTON) PROTOCOL

Protocol UNIFI-KC works as follows: First the each player should add to his private subset are Csk, m fake item sets, in order to hide its size. Then, finally the players jointly to compute the encryption of their private subsets are applying on those subsets to this encryption, where each player adds his turn and own layer of encryption using his private secret key. At the end of the stage, every item set in each subset are to be encrypted by all of the players; the usage of a commutative encryption scheme to ensures that all item sets are, eventually, encrypted item sets also to be used. Finally compute the union of those subsets in their encrypted of form. Finally, they decrypted the union set and remove from the item sets which are identified as fake.

STEPS FOR SECURE COMPUTATIONS OF ALL ITEMS SET (BYK&C):

1. Cryptographic Primitive Selection.

- Player are to be selects the needed commutative cipher and corresponding private key
- Player will be selecting hash function to apply on all of the item sets prior to encryption

- Player are to be compute lookup table with hash values to find reimages of the given hash values

2. All item sets to be encrypted.

3. Item sets are to be Merging.

- Each odd player to sends his encrypted set to the player 1.
- Each even player to sends his encrypted set to the player
- Player 1 unifies all of the item sets that were sent by the
- Player 2 unifies all of the item sets that were sent by the even players and removes duplicates.
- Player 2 sends his permuted list of the item sets to the Player 1.
- Player 1 unifies his list of the item sets and the lists received from the Player 2 and then finally remove duplicates from the unified list. Denoting the final list by ECSK.

4. Decryption.

B. SECURE MULTIPARTY PROTOCOL FOR COMPUTING THE OR OF PRIVATE BINARY VECTORS THRESHOLD PROTOCOL

In this module multiparty protocol is constructed by using the novel approach called the threshold protocol. It works by finding the binary vector of all the local item sets that were identified previously. The procedure of this threshold protocol is defined in the detailed manner as follows,

ALGORITHM:

Input: Each player P_m has an input binary vector $b_m \epsilon Z_2^n$, $1 \leq m \leq M$

Output: $b:=T_t(b_1,\ldots b_M)$

1. Each P_m selects M random share vectors $b_{m,l} \epsilon Z_{M+1}^n$, $1 \leq l \leq M$, such that $\sum_{l=1}^{M} b_{m,l} = b_m mod(M+1)$
2. Each Pm sends $b_{m,l}$ to P_l for all $1 \leq l \neq m \leq M$
3. Each Pl computes $s_l = (s_l(1),\ldots s_l(n)) := \sum_{m=1}^{M} b_{m,l} mod(M+1)$
4. Players P_l, $2 \leq l \leq M-1$, send s_l to P1
5. P_1 computes $s = (s_l(1),\ldots s_l(n)) := \sum_{l=1}^{M-1} s_l mod(M+1)$
6. For i=1,…n do
   a. If $(s(i)+s_m(i)) \bmod (M+1) < t$ set b(i) = 0
   b. Otherwise set b(i) = 1
7. End for
8. Output b=(b(1),…,b(n))

C. IMPROVED PROTOCOL FOR THE SECURE COMPUTATION OF ALL LOCALLY FREQUENT ITEM SETS

This is 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 $F^{k-1}{}_s$

ALGORITHM:

Input: Each player P_m has a subset $C_s^{k,m} \subseteq Ap\left(F_s^{k-1}\right), 1 \leq m \leq M$

Output: $C_s^k = m = 1MC_s^{k,m}$

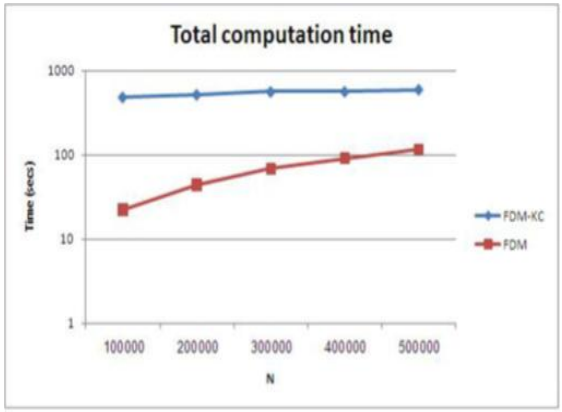
1. Each player P_m encodes his subset $C_s^{k,m}$ as a binary vector b_m of length $n_k = \left|Ap\left(F_s^{k-1}\right)\right|$, in accord with the agreed ordering $Ap\left(F_s^{k-1}\right)$
2. The players invoke protocol THRESHOLD-C to compute $b=T_1(b_1,\ldots b_M)=m = 1Mb_m$
3. $C_s^{k,m}$ is the sub set of $Ap\left(F_s^{k-1}\right)$ that is described by b

D. IDENTIFYING ALGORITHM THE GLOBALLY S-FREQUENT ITEM SETS

Protocols UNIFI-KC and UNIFI yield the set C^ks that consists of all item sets that are locally s-frequent in at smallest one site. Those are the k-item sets that have a potential to be also globally s-frequent. That computation must not reveal the local support in one of the sites. Agreement x be one of the candidate item sets in C^ks. Then x is globally s-frequent if and only if

$$\Delta(x) := supp(x) - sN = \sum_{m=1}^{M} (supp_m(x) - sN_m) \geq 0$$

We describe here the solution that was proposed by Kantarcioglu and Clifton. They considered two possible settings. If the necessary output contains all of the globally s-frequent item sets, as well as the magnitudes of their supports, then the values of D(x) can be revealed for all x $\in$ C^ks. where the private addend of



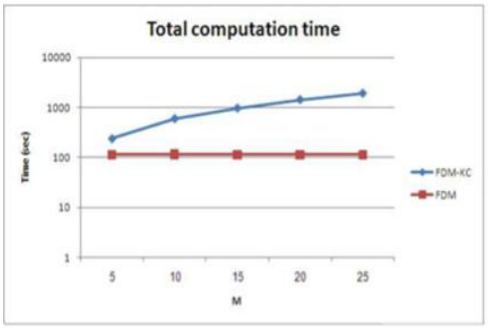
Pm is suppm(x)-sNm..

E. IDENTIFYING ALL (S, C)-ASSOCIATION RULES

Once the set Fs of all s-frequent item sets is found, we may proceed to look for all (s; c)-Association rules (rules with support at least sN and confidence at least c), For X, Y ϵ Fs, where X $\cap$Y = $\emptyset$, the corresponding association rule X $\Rightarrow$ Y has confidence at least c if and only if supp(X $\cup$ Y)/supp(X) $\geq$ c, or, equivalently,

$$C_{X,Y} := \sum_{m=1}^{M} (supp_m(X \cup Y) - c \cdot supp_m(X)) \geq 0$$

Since |CX;Y| $\leq$ N, then by taking q = 2N + 1, the players can verify inequality, in parallel, for all candidate association rules.



F. IDENTIFYING ASSOCIATION RULES OVER VERTICALLY PARTITIONED DATABASES:

Let Site1 and Site2 are two sites possessing vertically partitioned databases DB1 and DB2. Site1 has L number of attributes and Site2 has M number of attributes. Let MinSup be the support threshold specified by the user, and n be the total number of communications. So the total number of elements for the two databases is L M, where Site1 has attributes A1 through AL and Site2 has the remaining M attributes B1 through BM. Transactions are for the two databases consists of values of zero's or one's for L M attributes. Let *Xi* and *Yi* are vectors represent columns in the database that is ix i-1, if and only if row I has value 1 for attribute X. The scalar product of two cardinality n vectors *X* and *Y* is defined as

$$\vec{X}.\vec{Y} = \sum_{i=1}^{n} X_i.Y_i$$

Every site communicates with its successor site only except the last site, Site n communicates with DM. DM is having communications with all sites, Site1 to Site n. Each site performs the computations by using scalar product concept with its own computed results and the computed results obtained from its predecessor site.

EXAMPLE:

Let D be a database of N = 18 item sets over a set of L = 7 items, A = {1, 2, 3, 4, 5,6,7}. It is partitioned between M = 3 players and the corresponding partial databases are:

D1 = {17, 12345, 124, 1245, 16, 195, 239, 24, 24}
D2 = {1234, 138, 27, 234, 2645}
D3 = {1254, 128, 184, 33}.

For example, D1 includes N1 = 9 transactions, the third of Min-Support (s) or threshold support ($0 < s \leq 1$) = 0.43 Minimum Confidence (c) or threshold Confidence ($0 < c \leq 1$) = 0.43.

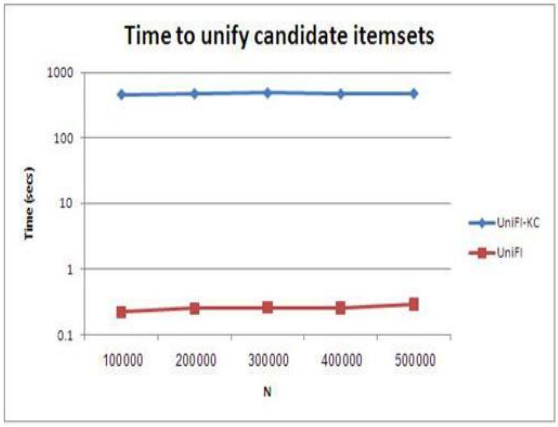
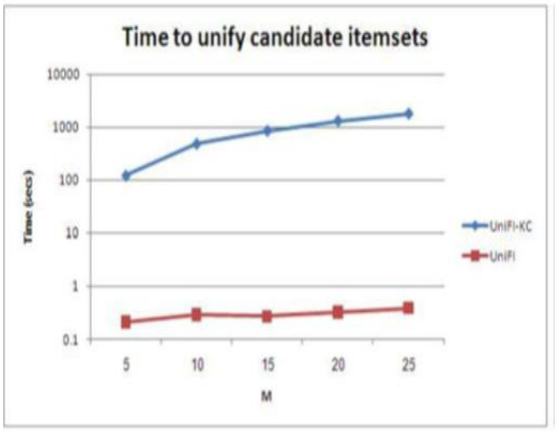


Figure 1: Computational and communication costs versus the number of transactions N

Comparatively study of FDM and FDM-KC is shown in the figure 1 and figure 2.Here in the figure 1 two graphs shows the number of transaction N has little effect on the runtime of FDM-KC. The second set of graph in figure 2 illustrates how the Computation and communication costs versus the number of players *M*

From the first set of the experiments, we can see that N has little effect on the runtime of the unification protocols, UNIFI-KC and UNIFI, nor on the bit communication cost. However, since the time to identify



the globally s-frequent item sets does grow linearly with N, and that procedure is carried out in the same manner in FDM-KC and FDM, the advantage of Protocol FDM over FDM-KC in terms of runtime decreases with N.While for N ¼ 100;000, Protocol FDM is 22 times faster than Protocol FDM-KC, for N ¼ 500;000 it is five times faster.

The second set of experiments shows how the computation and communication costs increase with M. Finally, the third set of experiments shows that higher support thresholds entail smaller computation and communication costs since the number of frequent item sets decreases.

Fig.2. Computational and communication costs versus the number of players M.

RELATED WORK

Previous work in privacy preserving data mining has considered two associated settings. One, in which the data owner and the data miner are two changed entities, and alternative, in which the data distributed among several parties who discussed secure clustering using the EM algorithm over vertically distributed data. The difficult of distributed association rule mining was studied in [1], [3], [6] in the vertically setting are, where each party holds a different set of attributes, and in [8] in the horizontal setting. Also the work of [9] considered this problem in the horizontal set- ting, but they considered large-scale systems in which, on top of the revelries that hold the data records (resources) there are also managers which are computers that assist the resources to decrypt messages; another assumption made in [9] that distinguishes it from [8] and the present study is that no collusions occur between the different network nodes properties or managers. The difficult of secure multiparty computation of the union of private sets was studied in [7], [5], [3], as well as in [8].

Freedman et al. [4] present a privacy-preserving protocol for set joints. It may be used to calculate also set unions through set complements, since A[B ¼ A\B. Kissner and Song [2] present a method for representing sets of polynomials, and finally give several privacy-preserving protocols for set operations using these demonstrations. They consider the threshold set blending problem, which is closely interrelated to the threshold function. The communication in the clouds of the results in those two works, as well as in [8]'s and in our solutions, depends linearly on the size of the ground set. However, as the protocols in [4], [2] use homomorphic encryption, while that of [8] uses commutative encryption of their computational costs stay significantly higher than ours. The work of Brickell and Shmatikov [7] is an omission, as their solution demands a

communication overhead that is logarithmic in the size of the pounded set.

However, they painstaking only the case of two players and the logarithmic communication in the clouds occurs only when the size of the intersection of the two sets is bounded by a perpetual. The privacy necessities are that the server gets no information about w and that the client gets no information about other pairs in the server's database. If we take all pi to be the unfilled string, then the only data to the client gets is whether or not w is in the set fx1;...;xng. Hence, in that case the privacy-preserving keyword search problem reduces to the set addition problem. Another solution for the set addition problem was recently proposed in [3], using a protocol for oblivious polynomial evaluation.

CONCLUSION

Fast Distributed Mining (FDM) algorithm is used to fetch frequent item sets in vertically Distributed Databases. Secure multiparty algorithms are used to mine privacy preserved frequent patterns from different databases. Unifying lists of locally Frequent Item sets (UNIFI-KC) algorithm is integrated with threshold function and set inclusive functions. The system is enhanced to mine rules under vertically distributed environment. The system supports vertically partition based rule mining of the process. Communication cost and computational cost and load is reduced in the distributed rule mining of the process. Sensitive attributes and item sets are protected by the system. The system improves the rule mining accuracy level on.

REFERENCES

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

[2]. Freedman.M, Ishai.Y, Pinkas.B, and Reingold.O, "Keyword Search and Oblivious Pseudorandom Functions," Proc. Second Int'l Conf. Theory of Cryptography (TCC), pp. 303-324, 2005.

[3]. Jiang.W and Clifton.C, "A Secure Distributed Framework for Achieving k-Anonymity," The VLDB J., vol. 15, pp. 316-333, 2006.

[4]. Kantarcioglu.M ,Nix.R, and Vaidya.J, "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.

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

[7]. Brickell.J and Shmatikov.V, "Privacy-Preserving Graph Algorithms in the Semi-Honest Model," Proc. 11th Int'l Conf. Theory and Application of Cryptology and Information Security (ASIACRYPT), pp. 236-252, 2005.

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

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

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