INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 3 (2018) PAGES 3916 - 3925
RECEIVED : 15.05.2018 PUBLISHED : 15.06.2018
June 15, 2018

FAST DISTRIBUTED MINING (FDM)OF ASSOCIATION RULES IN HORIZONTALLY DISTRIBUTED DATABASES

[1]K.Sureshkumar, ME Scholar, Department Of Computer Science and Engineering,Vmkv Engineering College, Salem.
[2]Dr. Nithya.,B.E.,M.E.,Ph.D Professor & Head of the DepartmentAssistant professor, Department of Computer Science and Engineering, Vmkv engineering college, Salem.
[3]S. Senthil Kumar , Assistant professor, Department of Computer Science and Engineering,Vmkv Engineering college, Salem.

ABSTRACT

We propose a protocol for secure mining of association rules in horizontally distributed databases. Our protocol, like theirs, is based on the Fast Distributed Mining (FDM) algorithm which is an unsecured distributed version of the Apriori algorithm. The main ingredients in our 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. Our protocol offers enhanced privacy with respect to the protocol. In addition, it is simpler and is significant more efficient in terms of communication rounds, communication cost and computational cost.

1. INTRODUCTION

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, x1, . . . , xM, andthey wish to securely compute y = f(x1, . . . , xM) for some public function f. If there existed atrusted third party, the players could surrender to him their inputs and he would perform thefunction evaluation and send to them the resulting output. In the absence of such a trusted thirdparty, it is needed to devise a protocol that the players can run on their own in order to arrive atthe required output y. Such a protocol is considered perfectly secure if no player can learn fromhis view of the protocol more than what he would have learnt in the idealized setting where thecomputation is carried out by a trusted third party. Yao was the first to propose a generic solutionfor this problem in the case of two players. Other generic solutions, for the multi-party case, were later proposed in than the given thresholds s and c, respectively. As the above mentionedgeneric solutions rely upon a description of the function f as a Boolean circuit, they can beapplied only to small inputs and functions which are realizable by simple circuits. In morecomplex settings, such as ours, other methods are required for carrying out this computation. Insuch cases, some

relaxations of the notion of perfect security might be inevitable when lookingfor practical protocols, provided that the excess information is deemed benign (see examples ofsuch protocols. 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. (The private subset of a given player, as we explain below, includes the item sets that are s-frequent in his partial database.) That is the costliest part of the protocol and its implementation relies upon cryptographic primitives such as commutative encryption, oblivious transfer, and hash functions. 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. While such leakage of information renders the protocol not perfectly secure, the perimeter of the excess information explicitly bounded in and it is argued there that such information leakage is innocuous, whence acceptable from a practical point of view Herein we propose an alternative protocol for the secure computation of the union of private subsets. The proposed protocol improves upon that in in terms of simplicity and efficiency as well as privacy. In particular, our protocol does not depend on commutative encryption and oblivious transfer (what simplifies it significantly and contributes towards much reduced communication and computational costs). While our solution is still not perfectly secure, it leaks excess information only to a small number (three) of possible coalitions, unlike the protocol of that discloses information also to some single players. In addition, we claim that the excess information that our protocol may leak is less sensitive than the excess information leaked by the protocol. The protocol that we propose here computes a parameterized family of functions, which we call threshold functions, in which the two extreme cases correspond to the problems of computing the union and intersection of private subsets. Those are in fact general-purpose protocols that can be used in other contexts as well. Another problem of secure multiparty computation that we solve here as part of our discussion is the set inclusion problem; namely, the problem where Alice holds a private subset of some ground set, and Bob holds an element in the ground set, and they wish to determine whether Bob's element is within Alice's subset, without revealing to either of them information about the other party's input beyond the above described inclusion.

2. RELATED WORK

Previous work in privacy preserving data mining has considered two related settings. One, in which the data owner and the data miner are two different entities, and another,in which the data is distributed among several parties whodiscussed secure clustering using the EM algorithm over horizontally distributed data. The problem of distributed association rule mining was studied in in the vertical setting, where each party holds a different set of attributes, and in the horizontal setting. Also the work of considered this problem in the horizontal setting, but they considered large-scale systems in which, on top of the parties that hold the data records (resources) thereare also managers which are computers that assist theresources to decrypt messages; another assumption made in that distinguishes it from and the present study is that no collusions occur between the different network nodes—resources or managers.

3. LITERATURE SURVEY

Association rule mining and frequent item set mining are two popular and widely studied data analysis techniques for a range of applications. In this paper, we focus on privacy-preserving mining on vertically partitioned databases. In such a scenario, data owners wish to learn the association rules or frequent item sets from a collective data set and disclose as little information about their (sensitive) raw data as possible to other data owners and third parties. To ensure data privacy, we design an efficient homomorphic encryption scheme and a secure comparison scheme. We then propose a cloud-aided frequent item set mining solution, which is used to build an association rule mining solution. Our solutions are designed for outsourced databases that allow multiple data owners to efficiently share their data securely without compromising on data privacy.

Our solutions leak less information about the raw data than most existing solutions. In comparison to the only known solution achieving a similar privacy level as our proposed solutions, the performance of our proposed solutions is three to five orders of magnitude higher. Based on our experiment findings using different parameters and data sets, we demonstrate that the run time in each of our solutions is only one order higher than that in the best non-privacy-preserving data mining algorithms. Since both data and computing work are outsourced to the cloud servers, the resource consumption at the data owner end is very low.

Distributed data mining techniques are widely used for many applications viz; marketing, decisionmaking, statistical analysis etc. In distributed data environment, each of the involving sites contains local information which will be collaborated to extract global mining result. However, these techniques have been investigated in terms of privacy and security concerns of individual site's information.

4. EXISTING SYSTEM:

In Existing System, 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 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.

DISADVANTAGE:

- Less number of features in previous system.
- Difficulty to get accurate item set.

5. PROBLEM STATEMENT

The proposed protocol improves upon that interms of simplicity and efficiency as well asprivacy. Our protocol does not depend oncommutative encryption and oblivious transfer.We propose here computes a

parameterizedfamily of functions, which we call doorstepfunctions, in which the two great casescommunicate to the problems of computing theunion and intersection of private subsets.The excess information that our protocol may leakis less sensitive than the excess information leakedby the protocol This project proposd two novelsecure multi-party algorithms — one thatcomputes the union of private subsets that each ofthe interacting players hold and another that teststhe inclusion of an element held by one player in asubset held by another. The problem of securemultiparty computation that we solve here is theset inclusion problem.

Objective: We proposed a protocol for securemining of association rules in horizontallydistributed databases that improves significantlyupon the current leading protocol in terms ofprivacy and efficiency. The main ingredient inour proposed protocol is a novel secure multipartyprotocol for compute the union (orintersection) of private subsets that each of theinteracting players holds.

6. PROPOSED SYSTEM:

In Proposed System, propose an alternative protocol for the secure computation of the union of private subsets. The proposed protocol improves upon that in terms of simplicity and efficiency as well as privacy. In particular, our protocol does not depend on commutative encryption and oblivious transfer (what simplexes it significantly and contributes towards much reduced communication and computational costs). While our solution is still not perfectly secure, it leaks excess information only to a small number (three) of possible coalitions, unlike the protocol of that discloses information also to some single players. In addition, we claim that the excess information that our protocol may leak is less sensitive than the excess information leaked by the protocol.

MODULES:

1. **USER MODULE.**
2. **ADMIN MODULE.**
3. **ASSOCIATION RULE.**
4. **APRIORI ALGORITHM.**

MODULES DESCRIPTION:

ADMIN MODULE:

In this module, is used to view user details admin to view the item set based on the user processing details using association role with Apriori algorithm.

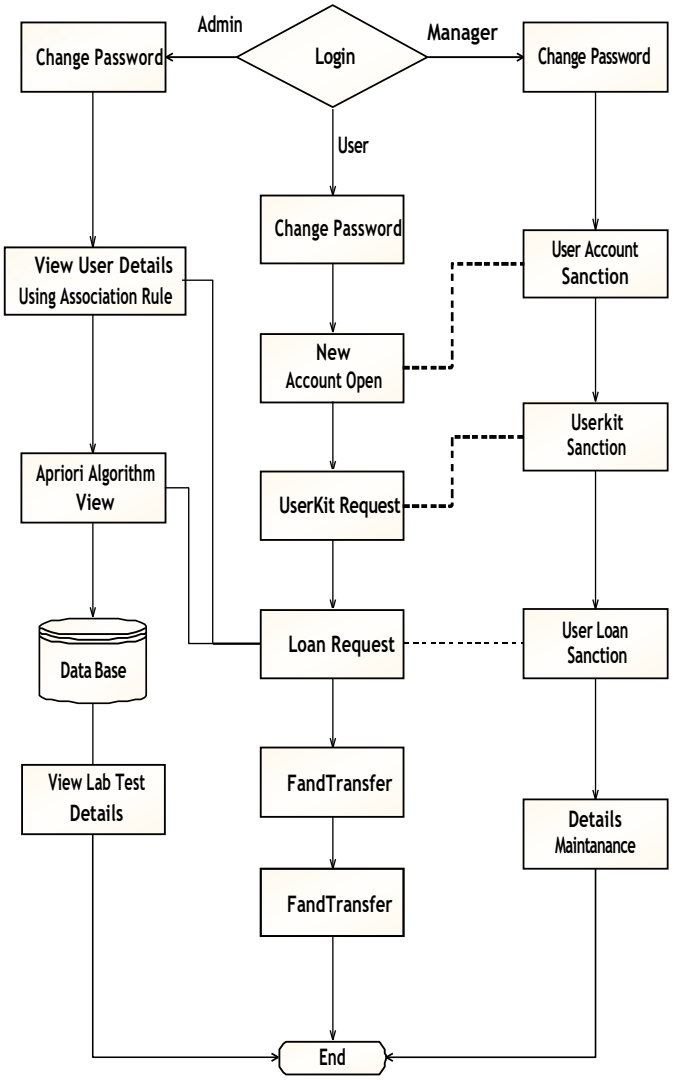
INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 3 (2018) PAGES 3916 - 3925
RECEIVED : 15.05.2018 PUBLISHED : 15.06.2018
June 15, 2018

USERMODULE:

In this module, privacy preserving data mining has considered two related settings. One, in which the data owner and the data miner are two different entities, and another, in which the data is distributed among several parties who aim to jointly perform data mining on the unified corpus of data that they hold.

In the first setting, the goal is to protect the data records from the data miner. Hence, the data owner aims at anonymizing the data prior to its release. The main approach in this context is to apply data perturbation. He perturbed data can be used to infer

DATAFLOW DIAGRAM



general trends in the data, without revealing original record information.

In the second setting, the goal is to perform data miningwhile protecting the data records of each of the data ownersfrom the other data owners.

ASSOCIATION RULE:

Association rules are if/then statements that help uncover relationships between seemingly unrelated data in a relational database or other information repository. An example of an association rule would be "If a customer buys a dozen eggs, he is 80% likely to also purchase milk."

Association rules are created by analyzing data for frequent if/then patterns and using the criteria support and confidence to identify the most important relationships.Support is an indication of how frequently the items appear in the database. Confidence indicates the number of times the if/then statements have been found to be true.

APRIORI ALGORITHM:

Apriori is designed to operate on databases containing transactions. The purpose of the Apriori Algorithm is to find associations between different sets of data. It is sometimes referred to as "Market Basket Analysis". Each set of data has a number of items and is called a transaction. The output of Apriori is sets of rules that tell us how often items are contained in sets of data.

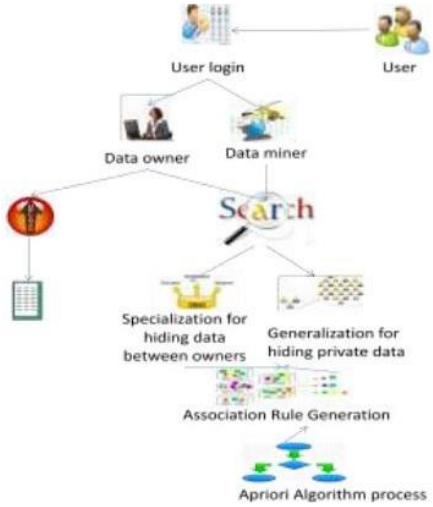
ALGORITHM - FAST DISTRIBUTED MINING (FDM)

The FDM algorithm proceeds as follows:

(1) Initialization
(2) Candidate Sets Generation
(3) Local Pruning
(4) Unifying the candidate item sets
(5) Computing local supports
(6) Broadcast Mining Results

INTERNATIONAL RESEARCH JOURNAL IN ADVANCED ENGINEERING AND TECHNOLOGY (IRJAET) E - ISSN: 2454-4752 P - ISSN : 2454-4744
VOL 4 ISSUE 3 (2018) PAGES 3916 - 3925
RECEIVED : 15.05.2018 PUBLISHED : 15.06.2018
June 15, 2018

7. SYSTEM ARCHITECTURE



The above system architecture explains the usermodule which enlists the privacy preserving datamining has considered two connected settings.One, in which the data owner and the data minerare two different individual, and another, in whichthe data is scattered among several parties whoaim to jointly perform data mining on the unifiedcorpus of data that they grip In the first location,the goal is to protect the data records from thedata miner. Hence, the information holder aims atanonymizing the data prior to its release. Themain approach in this framework is to apply dataperturbation. He disconcerted data can be used toinfer general trends in the data, withoutinformative original documents information. Inthe second setting, the goal is to perform datamining while protecting the data records of eachof the data owners from the other data owners.

The work of the administratoris to view userdetails. direction to view the item set based on theuser processing details using association role withApriori algorithm connection rules are if/thenstatements that help uncover relationshipsbetween seemingly unrelated data in a relationaldatabase or other information depository Anexample of an association rule would be "If acustomer buys a dozen seed, he is 80% likely toalso purchase develop Association set of laws arecreated by analyzing data for frequent if/thenpatterns and using the criteria support andconfidence to identify the most importantassociations support is an indication of howregularly the items appear in the database. Self-confidenceindicate the number of period theif/then statement have been found to be true.

ALGORITHM

1) L1 = flarge 1-itemsetsg;
2) for (k = 2; Lk1 6= ;; k++) do begin
3) Ck = apriori-gen(Lk1); // New candidates
4) forall transactions t 2 D do begin

5) Ct = subset(Ck , t); // Candidates contained in t
6) forall candidates c 2 Ct do
7) c:count++
8) end
9) Lk = fc 2 Ck j c:count _ minsupg
10) end
11) Answer = Sk Lk;

LK- Set of large k-itemsets(those with minimum
support).Each member of this set hastwo _elds itemset and ii) supportcount.Set of candidate k-itemsets
Ct- (potentially large itemsets).Each member of this set hastw:i) itemset and ii) supportcount.
Ck - Set of candidate k-itemsetswhen the TIof the generating transactionsarekept.

8. EXPERIMENTAL RESULTS

We compared the performance of two secureimplementations of the FDM algorithm. In thefirst implementation(denoted FDM-KC), weexecuted the unification step (Step 4 in FDM)using procedure UNIFI-KC, where thecommutative nobody was 1024-bit RSA in thesecond implementation (denoted FDM) we usedour procedure UNIFI, where the keyed-hashpurpose was HMAC. In both implementations, we implemented in FDM algorithm.In the securemanner that was described. We tested the twoimplementations with respect to three measures:
1) Total working out time of the completeprotocols (FDMKCand FDM) over all players. That measure includethe Apriori computation time[8][9] , and the time toidentify the globally *s*-frequent itemsets.
2) Total working out time of the unificationprotocols only.
3. (The latter two procedures are implement in the
same way in both Protocols FDM-KC and FDM.)
(UNIFI-KC and UNIFI) over all players.

4) Total message size. We ran three experimentsets, where each set tested theDependence of the above measures on a diverseparameter:
• N — the number of transactions in the unifieddatabase,
• M — the number of players, and
• s — the threshold support size.

In our basic configuration, we took N = 500, 000,
M = 10 and s = 0.1. In the first experiment set, we
kept M and s fixedand tested several values of N. In the second experiment set,we kept N and s fixed and varied$M$. In the third set, we kept$N$ and M fixed and varied s. The results in each ofthose experiment sets are shown above. All experiments were implemented in C# (.net 4) and were executed on an Intel(R) Core(TM)i7-2620Mpersonal computer with a 2.7GHz CPU, 8 GB of RAM, and the 64-bit operating system Windows 7Professional SP1.

CONCLUSION

We proposed a protocol for secure mining of association rules in horizontally distributed databases that improves significantly upon the current leading protocol in terms of privacy and efficiency. One of the main ingredients in our proposed protocol is a novel secure multi-party protocol for computing the union (or intersection) of private subsets that each of the interacting players holds. Another ingredient is a protocol that tests the inclusion of an element held by one player in a subset held by another. Those protocols exploit the fact that the underlying problem is of interest only when the number of players is greater than two.

One research problem this studies an efficient protocol for inequality verifications that uses the existence of a semi honest third party. Such a protocol might enable to further improve upon the communication and computational costs of the second and third stages of the protocol. Other research problems that this study suggests is the implementation of the techniques presented here to the problem of distributed association rule mining in the vertical setting the problem of mining generalized association rules, and the problem of subgroup discovery in horizontally partitioned data.

REFERENCES

[1] T. Brijs, G. Swinnen, K. Vanhoof, and G. Wets, “Using association rules for product assortment decisions: A case study,” in SIGKDD 1999.

[2] S. E. Brossette, A. P. Sprague, J. M. Hardin, K. B. Waites, W. T. Jones, and S. A. Moser, “Association rules and data mining in hospital infection control and public health surveillance,” Journal of the American medical informatics association, vol. 5, no. 4, pp. 373–381, 1998.

[3] B. Mobasher, H. Dai, T. Luo, and M. Nakagawa, “Effective personal-ization based on association rule discovery from web usage data,” in WIDM 2001.

[4] C. Creighton and S. Hanash, “Mining gene expression databases for association rules,” Bioinformatics, vol. 19, no. 1, pp. 79–86, 2003.

[5] X. Yin and J. Han, “Cpar: classification based on predictive association rules.” in SIAM SDM 2003.

[6] R. Agrawal, R. Srikant et al., “Fast algorithms for mining association rules,” in VLDB 1994.

[7] M. J. Zaki, “Scalable algorithms for association mining,” IEEE Transac-tions on Knowledge and Data Engineering, vol. 12, no. 3, pp. 372–390, 2000.

[8] J. Han, J. Pei, and Y. Yin, “Mining frequent patterns without candidate generation,” in ACM SIGMOD 2000.

[9] J. Vaidya and C. Clifton, “Privacy preserving association rule mining in vertically partitioned data,” in SIGKDD 2002.

[10] M. Kantarcioglu and C. Clifton, "Privacy-preserving distributed mining of association rules on horizontally partitioned data," IEEE Transactions on Knowledge and Data Engineering, vol. 16, no. 9, pp. 1026–1037, 2004.