International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 05, March 2016
Available at http://internationaljournalofresearch.org

Privacy Preserved Mining of Association Rules Over Horizontally Partitioned Data

[1] Santhi.Siruvuri & Dr.A.Jagan[2]
[1] M.Tech Student, Department of CSE, B.V. Raju Institute of Technology, Medak, Telangana, India.
[2] Head of the Department, Department of CSE, B.V. Raju Institute of Technology, Medak, Telangana, India.
[1]siruvurisanthi@gmail.com; [2]jagan.amgoth@bvrit.ac.in;

ABSTRACT:
In this paper, we survey the basic fundamental paradigms and notions of secure multiparty computation to the field of privacy-preserving data mining. Additionally reviewing definitions and constructions for secure multiparty computation, we discuss the issue of efficiency and demonstrate the difficulties involved in constructing highly efficient protocols. The main ingredients in the protocol are two new secure multi-party algorithms one who reckon the particular abutment connected with non-public subsets that every one of the interacting parties cling on and another that test the admittance of an element placed by one particular party in a very subset placed by another. We tend to analyze the performance of secure implementations of the efficient protocols.

Keywords: Data mining; multi party computation; association rules and privacy

1 INTRODUCTION

Data mining technology has emerged as a means of identifying patterns and trends from large quantities of data. Data mining and data warehousing go hand-in-hand: Most tools operate by gathering all data into a central site, then running an algorithm against that 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. Previous work inside privacy protecting data mining has regarded two similar settings. 1, in how the data owner along with the data miner are two different entities, as well as another, when the data will be distributed between several parties who try and jointly conduct data mining around the unified corpus involving data that they hold. From the first establishing, the goal is to protect the information records from your data miner. For this reason, the data owner is aimed at anonymizing the information prior to help its release. The key approach with this context is to apply data perturbation. The strategy is the perturbed data may be used to infer standard trends inside the data, without revealing original record information. In your second setting, the goal is to perform data mining although protecting the info records of each of the data owners from your other data owners. That is a problem involving secure multi-party calculation. The normal approach here's cryptographic instead of probabilistic.

This paper proposes an alternative protocol for your secure computation with the union involving private subsets. The offered protocol boosts upon that when it comes to simplicity as well as efficiency along with privacy. For example, our protocol does not depend upon commutative encryption as well as oblivious exchange what simplifies this significantly as well as contributes toward much lessened communication as well as computational costs. While the solution is not flawlessly secure, it

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 05, March 2016
Available at http://internationaljournalofresearch.org

leaks excess information merely to a small amount of possible coalitions, unlike the protocol of which discloses information and also to some individual players. Also, this paper declares that the unwanted information our protocol may perhaps leak will be less sensitive than the excess information leaked through the protocol. The protocol until this paper propose here computes the parameterized class of functions, that this papers call tolerance functions, when the two intense cases correspond to the complications of calculating the partnership and intersection involving private subsets. Those are in fact general-purpose protocols that can be used in various other contexts at the same time. Another problem of safe multiparty computation until this paper fix here as part of our discussion will be the set inclusion problem; that is, the problem where Alice holds a non-public subset involving some terrain set, and Bob holds a component in the set, and they wish to determine regardless of whether Bob's element is within Alice's subset, without exposing to either ones information around the other party's suggestions beyond the above mentioned described introduction.

2 RELATED WORK

Association rule mining has been used extensively for the classical problem of market basket analysis where it is required to find the buying habits of customers. Determining what products customers are likely to buy together can be very useful for planning and marketing. Association rules are used to show the relationships between these data items. Many centralized algorithms exist for Association Rule Mining (ARM). Most of the algorithms depend on the discovery of frequent Itemsets for generation of association rules. Since the total number of Itemsets is exponential in terms of the number of items, it is not possible to count the frequencies of these sets by reading the database in just one pass. Different algorithms for the discovery of association rules aim at reducing the number of passes by generating candidate sets, which are likely to be frequent Itemsets. They attempt to eliminate infrequent sets as early as possible.

3 PROBLEM DEFINITION

Security against semi-honest adversaries might be justified if the parties participating in the protocol are somewhat trusted (say, if they are different institutions or agencies that need to compute a function of some information that regulations prevent them from sharing). This level of security is also justified if we trust the participating parties at the time they execute the protocol, but suspect that at a later time an adversary might corrupt them and get hold of the transcript of the information received in the protocol.

Recent work, describes a awful optimized accomplishing of a two-party protocol that offer security against malicious adversaries of Lindell and Pinkas, but protocols concentrating on the using the same efficiency that never available for the multi-party case. Even that accomplishment introduces a substantial performance penalty, as since it must substantially increase the amount of inputs and the size of the circuit, to reckon multiple copies of the circuit. Given this issue in achieving security against malicious adversaries, the present version of FairplayMP handles solely the semi-honest case. [1]

Our alternative in the semi-honest model follows previous work towards privacy-preserving data processing like Lindell and Pinkas' construction for a privacy-preserving version in the ID3 decision tree learning algorithm for privacy-preserving classification.

The authors in [10], proposed a new algorithm for semi-honest model with negligible collision probability is a modified algorithm of privacy enhancing association rule mining on distributed homogenous dataset. Informally, security of a

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 05, March 2016
Available at http://internationaljournalofresearch.org

protocol in the SMC paradigm is defined as computational indistinguishability from some ideal functionality, in which a trusted third party accepts the parties' inputs and carries out the computation. The ideal functionality is thus secure by definition. The actual protocol is secure if the adversary's view in any protocol execution can be simulated by an efficient simulator who has access only to the ideal functionality, i.e., the actual protocol does not leak any information beyond what is given out by the ideal functionality.

4 PROPOSED SYSTEM

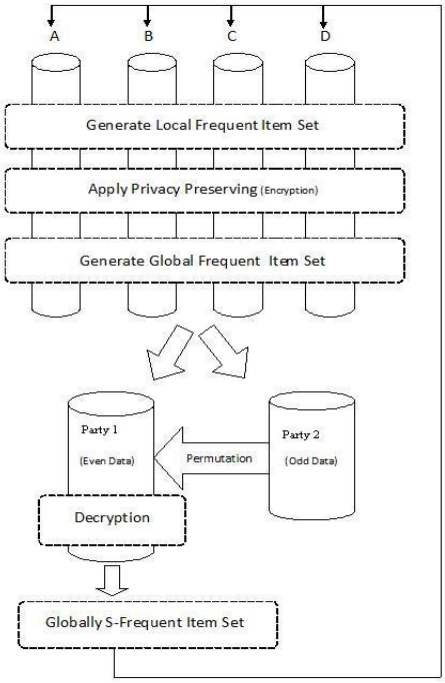

Fig 1: Architecture

A. Initialization

In this module let A be the set of items and D a transaction database (TDB) of transactions, each of which is a set of items. We denote the support of an itemset S ⊆ A as suppD(S) and the frequency by freqD(S). For each item i, suppD(i) and freqD(i) denote respectively the individual support and frequency of i. Let D denote the original TDB that the owner has. To protect the identification of individual items, the owner applies an encryption function to D and transforms it. The term item shall mean plain item by default. The notions of plain item sets, plain transactions, plain patterns, transaction over some set of items, A {a1; . . . ; aL} and each column represents one of the items in A. In other words, the (i,j) entry of D equals 1 if the ith transaction includes the item aj, and 0 otherwise. 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. 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. Each player selects a random private key and hash functions to apply to all item sets.

B. Candidate and Pruning

Each player Pm computes the set of all k-1 item sets that are locally frequent in his site and also globally frequent; namely, Pm computes the set locally as well as globally frequent sets. Then the Apriori algorithm is applied in order to generate the set B of candidate k-item sets. For each item in B, Pm computes suppm(X). Then retains only those item sets that are locally s-frequent. The collection of these item sets are denoted by C.

C. Encryption of Itemsets

In this module we introduce the encryption scheme, which transforms a TDB D into its encrypted version D. This scheme consists of

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 05, March 2016
Available at http://internationaljournalofresearch.org

main steps that use hashing techniques. All players compute a composite encryption of the hashed sets C. In first steps each player Pm hashes all item sets in C and then encrypts them using the key Km. Hashing is needed in order to prevent leakage of algebraic relations between item sets. Then, he adds to the resulting set faked item sets in order to hide the number of locally frequent item sets that he has. Then the players start a loop of M cycles, where in each cycle they perform the following operation: Player Pm sends a permutation of Xm to the next player Pm+1; Player Pm receives from Pm-1 a permutation of the set Xm-1 and then computes a new Xm as Xm. At the end of this loop, Pm holds an encryption of the hashed C using all M keys.

D. Merging Itemsets

In this module, the players merge the lists of encrypted item sets. At the completion of this stage P1 holds the union set C hashed and then encrypted by all encryption keys, together with some fake item sets that were used for the sake of hiding the sizes of the sets C those fake item sets are not needed anymore and will be removed after decryption in the next phase. The merging is done in two stages, where in the first stage the odd and even lists are merged separately. Not all lists are merged at once since if they were, then the player who did the merging would be able to identify all of his own encrypted item sets as he would get then from PM and then learn in which of the other sites they are also locally frequent. The merging is carried out according to steps below: Each odd player sends his encrypted set to players P1, Each even player sends his encrypted set to player P2, P1 unified all sets that were sent by the odd players and removes duplicates, P2 unifies all sets that were sent by the even players and removes duplicates, P2 sents its permuted list of itemsets to P1, P1 unifies its list of itemsets and list received from P2 and the removes duplicates from the unified list.

E. Decryption of Itemsets

In this module, a similar round of decryptions is initiated. At the end, the last player who performs the last decryption uses the lookup table T that was constructed in Step 4 in order to identify and remove the fake item sets and then to recover C. Finally, he broadcasts C to all his peers. For all playes, the last player decrypts all itemsets in encrypted C using corresponding K, and sends permuted and decrypted K to other player. It also decrypts all itemsets in EC, it also uses the lookup table to replace hash values with the actual itemsets and to identify and remove faked itemsets. The player finally broadcasts decrypted C

5. RESUTLS

The concept of this paper is implemented and different results are shown below, The proposed paper is implemented in Java technology on a Pentium-IV PC with minimum 20 GB hard-disk and 1GB RAM. The propose paper's concepts shows efficient results and has been efficiently tested on different Datasets.

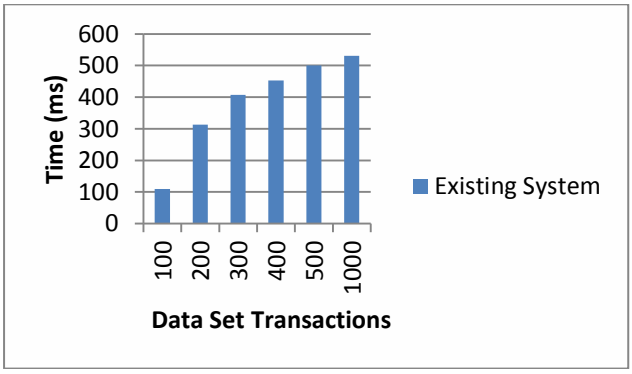

Fig. 2 Execution Time taken by Existing System.

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 05, March 2016
Available at http://internationaljournalofresearch.org

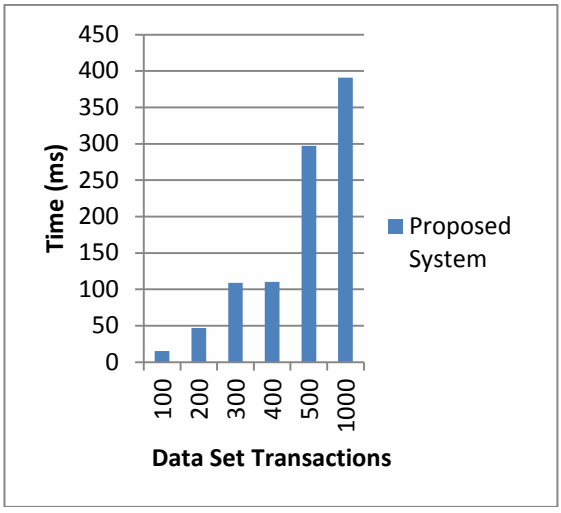

Fig. 3 Execution Time taken by Proposed System

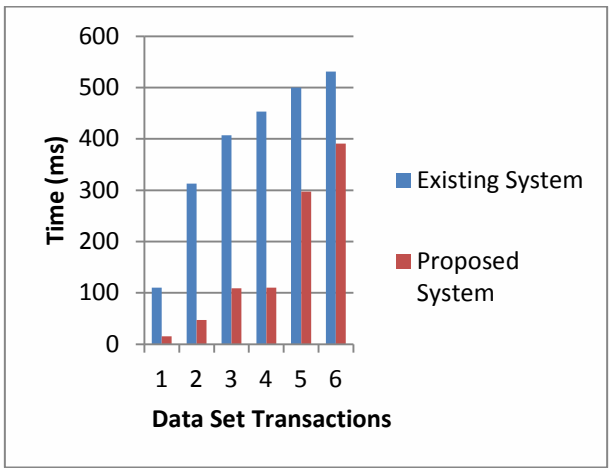

Fig 4: Analysis of computational time for the existing and proposed system

CONCLUSION

In this paper we tend to devise a protocol for reckoning association rules within a scenario of homogeneous data. The protocol is more efficient than current leading K and C protocol. Multi-party Computation being employed in big datasets that needs to preserve privacy of the private data with respect to various parties. Those practices exploit the truth that the main problem can be of interest only once the amount of parties are more than two. The future work is to devise an most efficient protocol for inequality verifications that uses the existence of semihonest third party and another in the implementation of techniques to the matter of distributed association rule mining in vertical setting.

REFERNCES

[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] 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] chin-chen chang, yu-chiang Li An Efficient Algorithm for Incremental Mining of association Rules *15t* (Tassa, 2014)*h international workshop IEEE (2005).*

[6] D.W. Cheung, V.T. Ng, A.W. Fu, Y. Fu, "Efficient mining of association rules in distributed databases", *IEEE Transactions on Knowledge and Data Engineering, vol 8, no 6, pp. 911-922, 1996.*

[7] G. Cormode and M. Garofalakis. Sketching probabilistic data streams. In SIGMOD '07: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, 2007.

[8] J.S. Park, M. Chen, P.S. Yu, "An effective hash-based algorithm for mining association rules", In Proceedings of ACM SIGMOD International Conference on Management of Data, San Jose, California, pp. 175-186, May 1995.

[9] L. Aouad, N. Khac, T. Kechadi, "Performance study of distributed Apriori-like

frequent item sets mining", Knowledge Information System, no 23, pp. 55-72, 2010.

[10] lberto Trombetta, Wei Jiang, Elisa Bertino, "Privacy Preserving Updates To Anonymous And Confidential Databases",*IEEE Transactions On Dependable And Secure Computing*, Vol. 8, No. 4, PP. 578-587, 2011.

[11] Murat Kantarcioglu, Chris Clifton, "Privacy-Preserving Distributed Mining Of Association Rules On Horizontally Partitioned Data", *Knowledge And Data Engineering,IEEE Transactions*,Vol. 16, No.9, 2004.

[12] Ramesh C. Agarwal, Charu C. Aggarwal, and V. V. V. Prasad. A tree projection algorithm for generation of frequent item sets. *Journal of Parallel and Distributed Computing, 61(3):350-371, 2001.*

[13] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. *In Proc. 20th Int. Conf. on Very Large Databases, pages 487–499, Santiago, Chile, 1994.*

[14] Ratchadaporn Amornchewin Probability-based Incremental Association Rules Discovery Algorithm with Hashing Technique. *[International Journal of Machine Learning and Computing, Vol.1, No. 1, April 2011.*

[15] Ratchadaporn Amornchewin, Worapoj Kreesuradej Incremental Association Rule Mining Using Promising Frequent Itemset Algorithm *[IEEE 2007]*.

[16] C. Clifton, M. Kantarcioglu, J. Vaidya, X. Lin, and M. Zhu, "Tools for privacy preserving distributed data mining", *ACM SIGKDD Explorations, 2003.*

[17]www.ics.uci.edu refers dataset.