RULE HIDING IN DISTRIBUTED DATABASE USING SECURE MINING.

Abhijit Vedpathak[1], Rohit Pawar[2], Mahesh Thorat[3], Siddharth Jadhav[4], Sanket Warkad[5]
[1] *Student, Computer Dept. SRESCOE Kopargaon, Maharashtra, India*
[2] *Student, Computer Dept. SRESCOE Kopargaon, Maharashtra, India*
[3] *Student, Computer Dept. SRESCOE Kopargaon, Maharashtra, India*
[4] *Student, Computer Dept. SRESCOE Kopargaon, Maharashtra, India*
[5] *Student, Computer Dept. SRESCOE Kopargaon, Maharashtra, India*

ABSTRACT

We are coming out with a complete detailed overview and classification of various approaches which have been applied to knowledge hiding of association rule mining. We are going to propose an FDM algorithm on centralized database to ensure the data quality at certain level. In order to maintain the quality of database less modification need to be done. We are going to use the sensitive association rules based on certain criteria. We describe the performance of the proposed algorithm and also analyse the result of existing algorithm for central data.

Keyword: - *Distributed Computation, Frequent Itemsets and Association Rule*

1. INTRODUCTION

We are going to implement a plan for secure mining of association rules in horizontally distributed databases. System is based on Apriori algorithm. 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.

With support and confidence, our goal is to find all association rules. In this system we protect individual transactions and global information. For this system, the inputs are the partial databases and the required output is the list of association rules that hold in the database with support and confidence.

1.1 Scope

1. The system which we are going to implement is based upon rules in horizontally distributed databases using secure mining.
2. The system is based on Apriori algorithm.
3. There are several sites that hold homogeneous databases, i.e., databases that share the same data but hold information on different sites.

1.2 Objective

1. To analyze data from horizontal distributed databases.
2. The goal is to perform data mining while protecting the data records of each of the owners from the other data owners.
3. To display frequent itemset from distributed system.
4. To find buyers behaviors by using frequent itemsets.
5. To protect the records from data miner.

2. RELEVANT THEORY

2.1 Existing System

In existing system, there are *N* users that hold private inputs *x*1, . . . , *xN*, and they wish to safely execute *y* = *f*(*x*1, . . . , *xN*) for some public function *f*. If there available a trusted third party, the users could surrender to him their inputs and he would evaluate the function and send to them the generated output. In the absence of a trusted third party, it is needed to setup a system that the users can run on their own in order to arrive at the required output *y*. Such a system is considered perfectly secured if no user can learn from his view of the system more than what he would have understand in the idealized setting where the computation is carried out by a trusted third party.

2.2 Proposed System

We propose an alternative rules 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. Our protocol does not depend on commutative encryption and oblivious. It leaks excess information only to a small number of possible block when our solution is still not perfectly secure, unlike the rule of that discloses information also to some single user. In addition, we say that the extra information that our protocol may leak is less sensitive than the extra information leaked by the protocol.

The system that we implement here computes a parameterized group of functions, which we call threshold functions, in which the two cases correspond to the problems of computing the union and intersection of private subsets.

3. SYSTEM ARCHITECTURE

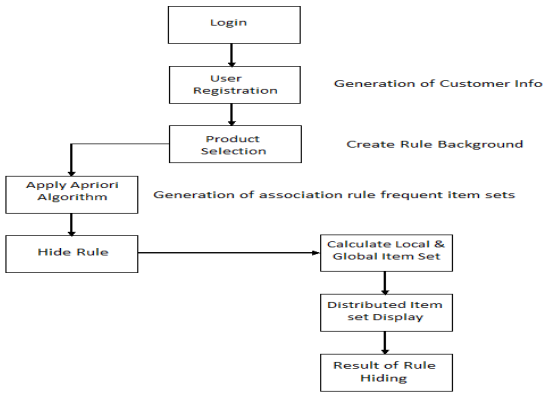

Fig-1: System Architecture.

Initially user needs to register on system. After registration the user is assign with unique key. This unique key contains all details of user who is logged in on that system. Suppose the web page the user is on contain n number of products. User select m numbers of product using check box. According to selected m products the code is generated in 0 & 1 format and this code is saved in text file. Now the Apriori Algorithm is applied on this text file. This same process is applied on all text files on other web sites. Using FDM (i.e., Fast Distributed Mining of association rules) Algorithm, all these text files from web sites are collected on Server.

3.1 Block Diagram

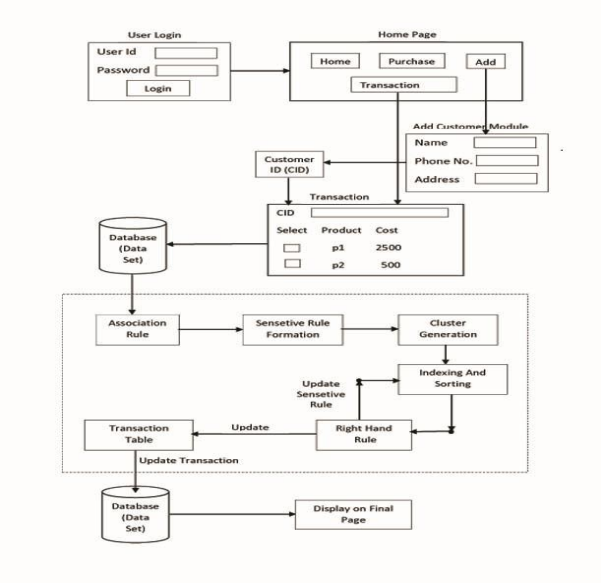

Fig-2: Block diagram of Rule hiding in distributed database using secure mining

Initially user needs to make new registration. When the registration is done successfully, the server will generate the unique key which will be assigned to the user. This unique key is send to our register email address for further references. This unique key will also be used by the website/server in order to identify the purchases of the particular product.

When the process of purchasing the product on the website is done, as per product selection the binary code is generated by the server. This generated binary code by the server is stored in database by using Apriori Algorithm and various other rules (Association rule, Right hand rule, etc.). This algorithm reverses the binary code in order to provide security from unauthorized user. After completion of all these process finally display the details of order.

4. IMPLEMENTATION MODULES

This section contains the details about the implementation modules.

4.1 Add customer Module

In this module we Registers the Customer and unique Customer ID are generated which is use in Consequent steps. Without registration customer cannot buy the items available on this site.

4.2 Association Rules Mining using Apriori Algorithm

Based on products selected by customer that is on transaction .We are going to apply Apriority Algorithm. Whichever products are selected to buy will be marked as 1 and which are not selected are marked as 0 and according to this buying entries are get added to database. This module also adds the association rule to the item set available on website.

4.3 Frequent Item sets Calculation

Frequent Item sets are calculated from Apriori. There are two possible settings, if required output includes all globally frequent item sets, as well as the length of their supports, then the values of $\Delta(x)$ can be discover for all. The more interesting setting, however, is the one where the support sizes are not part of the required output.

4.4 Distributed Item set Calculation

Frequent Item set on Distributed sites are Computed in This module**.** We compared the performance of two secure implementations of the FDM algorithm. In the first phase of implementation (FDM-KC), we executed the unification step using System UNIFI-KC, in which the commutative cipher was 1024-bit RSA in the second

implementation (denoted FDM) we used our System UNIFI, where the keyed-hash function was HMAC. We tested the two implementations with respect to three measures:

a) Total computation time of the complete systems (FDMKC and FDM) over all users. That measure includes the Apriori computation time, and the time to recognize the globally *s*-frequent itemsets, as described in later.
b) Total computation time of the unification systems (UNIFI-KC and UNIFI) only users.
c) Total message size. We ran three experiment on sets, where each and every set is tested the dependence of the mentioned measures on a different parameters: • N — the number of transactions in the unified database.

5. SYSTEM SPECIFICATIONS

This section contains various software as well as hardware specifications.

5.1 Software Specifications

5.1.1 Frontend Server

- JAVA
- TOMCAT Server

5.1.2 Backend Server

- SQL SERVER 2008

5.2 Hardware Specification

- 512 MB RAM
- PENTIUM-4, 2GHz

6. APPLICATION

1. We are going to implement a system for hiding the rules in distributed databases using secure mining that improves significantly upon the current leading system in terms of privacy and efficiency.

2. The main ingredient of the system is a novel secure multi-party system for computing the union (or intersection) of non public subsets that each of the interacting users holds.

7. CONCLUSION

We propose a system for hiding the rules in distributed databases using secure mining that improves privacy and efficiency. We propose the system for maximizing the privacy and minimizing information while sharing the data without disclosing individual's identity and secure the database owner privacy rules. For this model condensation are used for preserving covariance information and preserving privacy using Association rules which is based on modifying the database transaction.

REFERENCES

1. M. Kantarcioglu and C. Clifton. Privacy-preserving distributed mining of association rules on horizontally partitioned data. *IEEE Transactions on Knowledge and Data Engineering*, 16:1026–1037, 2004.
2. A.V. Evfimievski, R. Srikant, R. Agrawal, and J. Gehrke. Privacy preserving mining of association rules. In *KDD*, pages 217–228, 2002.
3. A. Schuster, R. Wolff, and B. Gilburd. Privacy-preserving association rule mining in large-scale distributed systems. In *CCGRID*, pages 411–418, 2004.
4. Tamir Tassa "Secure Mining of Association Rules in Horizontally Distributed Databases" IEEE
5. TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 26, NO. 4, APRIL 2014

6. 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.
7. D.W.L Cheung, V.T.Y. Ng, A.W.C. Fu, and Y. Fu, "Efficient Mining of Association Rules in Distributed Databases," IEEE Trans. Knowledge and Data Eng., vol. 8, no. 6, Dec. 1996.
8. R. Fagin, M. Naor, and P. Winkler, "Comparing Information without Leaking It," Comm. ACM, vol. 39, pp. 77-85, 1996.
9. M. Freedman, Y. Ishai, B. Pinkas, and O. Reingold, "Keyword Search and Oblivious Pseudorandom Functions," Proc. Second Int'l Conf. Theory of Cryptography (TCC), pp. 303-324, 2005.
10. M.J. Freedman, K. Nissim, and B. Pinkas, "Efficient Private Matching and Set Intersection," Proc. Int'l Conf. Theory and Applications of Cryptographic Techniques (EUROCRYPT), pp. 1-19-2004.