Analysis of Distributed Databases

Priyanka Pandey[1]& Javed Aktar Khan[2]
[1,2]Department of Computer Science and Engineering, Takshshila Institute of Engineering and Technology, Jabalpur, M.P., India
pandeypriyanka906@gmail.com; javedaktarkhan@takshshila.org

Abstract—
Data mining extract pattern/ knowledge from a large amount of database. In the applications that are based on the information sharing and additional challenges, when dealing with the data that containing sensitive or private information. There is no any common data mining techniques available that dealing with the private information without any leakage. There for the knowledge extracted from such data may disclose pattern with sensitive/ private information. This may put privacy on the individual/ group of parties. In the few last years, privacy preserving data mining has attracted the research interest and potential for wide area of applications. There are many techniques for privacy preservation like cryptography; anonymity and randomization etc. have been experimented for privacy preservation in data mining. In this paper we analyzed the different partitioning algorithm that is useful for partitioning the database in distributed environments.

Keywords: Database; Data Mining; Distributed Database; Partitioning Approach; Homogeneous Database; Heterogeneous Database.

1. Introduction

Due to the increased demand for knowledge discovery [1] [2] [3] in all industrial domains, it is necessary to store all the raw data and to provide useful patterns with respective to the user needs. Generally, the storage of all raw data will be done in a database maintained by respective organizations. Data mining techniques are available to retrieve useful information from large database. Prediction and description are the two fundamental goals of data mining. To full fill these goals many data mining techniques exist such as association rules, classification, clustering and so on. Among these, association rule has wide applications to discover interesting relationship among attributes in large databases [5] [6] [7]. Association rule mining is used to find the rules which satisfy the user specified minimum support and minimum confidence. In the process of finding association rules, the set of frequent item sets are computed as the first step and then association rules are generated based on these frequent item sets.

2. Database

A collection of related data, information and related pieces of data representing/capturing the information about a real-world enterprise or part of an enterprise. Collected and maintained to serve specific data management needs of the enterprise. Activities of the enterprise are supported by the database and continually update the database An Example University Database: Data about students, faculty, courses, research-laboratories, course registration/enrollment etc.

Two types of database environments exist namely centralized and distributed. In contrast to the centralized data base model, the distributed data base model assumes that the data base is partitioned into disjoint fragments and each fragment is assigned to one site. The issue of privacy arises when the data is distributed among multiple sites and no site owner wish to provide their private data to other sites but they are interested to know the global results obtained from the mining process.

The main aim in many distributed methods for privacy preserving data mining is to allow useful aggregate computations on the complete data set by preserving the privacy of the individual sites data/information.

3. Data Mining and Techniques

Data Mining is the analysis of (often large) observational data sets to find unsuspected relationships and to summarize the data in novel ways that are both understandable and useful to the owner [14]. There are many different data mining functionalities. A brief definition of each of these functionalities is now presented. The definitions are directly collated from [13]. Data characterization is the summarization of the general characteristics or features of a target class of data. Data Discrimination, on the other hand, is a comparison of the general features of target class data objects with the general features of objects from one or a set of contrasting classes. Association analysis is the discovery of association rules showing attribute value conditions that occur frequently together in a given set of data. Classification is the process of finding a set of models (or functions) that describe and distinguish data classes or concepts, for the purpose of being able to use the model to predict the class of objects whose class label is unknown. The derived model can be represented in various forms, such as classification rules, decision trees, mathematical formulae, or neural networks. Unlike classification and prediction, which analyze class-labeled data objects, clustering analyzes data objects without consulting a known class label. Outlier Analysis attempts to find outliers or anomalies in data. A detailed discussion of these various functionalities can be found in [13]. Even an overview of the representative algorithms developed for knowledge discovery is beyond the scope of this dissertation. The interested person is directed to the many books which amply cover this in detail [19],[13],[14].

4. Distributed Database

A distributed database is a database in which storage devices are not all attached to a common processing unit such as the CPU, controlled by a distributed database management system (together sometimes called a distributed database system). It may be stored in multiple computers, located in the same physical location; or may be dispersed over a network of interconnected computers. Unlike parallel systems, in which the processors are tightly coupled and constitute a single database system, a distributed database system consists of loosely-coupled sites that share no physical components. System administrators can distribute collections of data (e.g. in a database) across multiple physical locations. A distributed database can reside on network servers on the Internet, on corporate intranets or extranets, or on other company networks. Because they store data across multiple computers, distributed databases can improve performance at end-user worksites by allowing transactions to be processed on many machines, instead of being limited to one. Two processes ensure that the distributed databases remain up-to-date and current: replication and duplication.

4.1 Replication: It involves using specialized software that looks for changes in the distributive database. Once the changes have been identified, the replication process makes all the databases look the same. The replication process can be complex and time-consuming depending on the size and number of the distributed databases. This process can also require a lot of time and computer resources.

4.2 Duplication: It basically identifies one database as a master and then duplicates that database. The duplication process is normally done at a set time after hours. This is to ensure that each distributed location has the same data. In the duplication process, users may change only the master database. This ensures that local data will not be overwritten. Both replication and duplication can keep the data current in all distributive locations. Besides distributed database replication and fragmentation, there are many other distributed database design technologies. For example, local autonomy, synchronous and asynchronous distributed database technologies. These technologies' implementation can and does depend on the needs of the business and the sensitivity/confidentiality of the data stored in the database, and hence the price the business is willing to spend on ensuring data security, consistency and integrity. The classification of distributed database as homogeneous or heterogeneous database

1. **Homogeneous Distributed Database:** It has identical software and hardware running all databases instances, and may appear through a single interface as if it were a single database.
2. **Heterogeneous Distributed Database:** It has different hardware, operating systems, database management systems, and even data models for different databases.

5. Homogeneous DDBMS

In a homogeneous distributed database all sites have identical software and are aware of each other and agree to cooperate in processing user requests. Each site surrenders part of its autonomy in terms of right to change schema or software. A homogeneous DDBMS appears to the user as a single system. The homogeneous system is much easier to design and manage. The following conditions must be satisfied for homogeneous database:

1. The operating system used, at each location must be same or compatible.
2. The data structures used at each location must be same or compatible.
3. The database application (or DBMS) used at each location must be same or compatible.

6. Heterogeneous DDBMS

In a heterogeneous distributed database, different sites may use different schema and software. Difference in schema is a major problem for query processing and transaction processing. Sites may not be aware of each other and may provide only limited facilities for cooperation in transaction processing. In heterogeneous systems, different nodes may have different hardware & software and data structures at various nodes or locations are also incompatible. Different computers and operating systems, database applications or data models may be used at each of the locations. For example, one location may have the latest relational database management technology, while another location may store data using conventional files or old version of database management system. Similarly, one location may have the Windows NT operating system, while another may have UNIX. Heterogeneous systems are usually used when individual sites use their own hardware and software. On heterogeneous system, translations are required to allow communication between different sites (or DBMS). In this system, the users must be able to make requests in a database language at their local sites. Usually the SQL database language is used for this purpose. If the hardware is different, then the translation is straightforward, in which computer codes and word-length is changed. The heterogeneous system is often not technically or economically feasible. In this system, a user at one location may be able to read but not update the data at another location.

7. Vertical Partitioning

Vertical partitioning (a.k.a. heterogeneous distribution) of data implies that though different sites gather information about the same set of entities, they collect different feature sets. For example, financial transaction information is collected by banks, while the IRS collects tax information for everyone. An illustrative example of vertical partitioning and the kind of useful knowledge we can hope to extract is given in Figure 1. The figure describes two databases; one contains medical records of people while another contains cell phone information for the same set of people. Mining the joint global database might reveal information like Cell phones with Li/Ion batteries lead to brain tumors in diabetics.

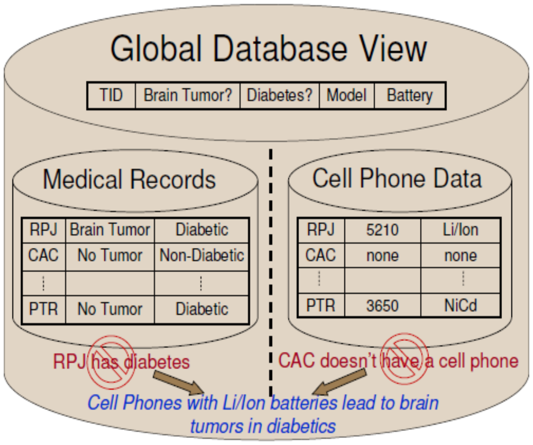

Figure1: Shows Vertical Partitioned Database

Unless otherwise stated, the model assumed is as follows:

There are k parties, P_0………. P_{k-1} . There are a total of n transactions for which information is collected. Party Pi collects information about m_i attributes, such that m = $\sum_{i=0}^{k-1} m_i$ is the total number of attributes/features. This thesis only considers privacy-preserving data mining in the case of vertical partitioning of data. For the sake of completeness, the following section gives some detail on horizontal partitioning of data.

8. Horizontal Partitioning

In horizontal partitioning (a.k.a. homogeneous distribution), different sites collect the same set of information, but about different entities. An example of that would be grocery shopping data collected by different supermarkets (also known as market-basket data in the data mining literature). Figure 2 illustrates horizontal partitioning and shows the credit card databases of two different (Local) credit unions. Taken together, one may find that fraudulent customers often have similar transaction histories, etc.

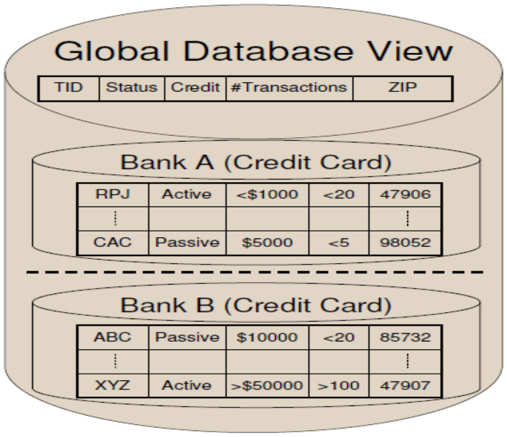

Figure 2: Shows Horizontal Partitioned Database

These different partitioning pose different problems, leading to different algorithms for privacy- reserving data mining.

9. Secure Multiparty Computation

Consider a set of parties who do not trust each other, or the channels by which they communicate. Still, the parties wish to correctly compute some common function of their local inputs, while keeping their local data as private as possible. This, in a nutshell, is the problem of Secure Multiparty Computation (SMC). It is clear that the problem we wish to solve, privacy-preserving data mining, is a special case of the secure multi-party computation problem. Before proposing algorithms that preserve privacy, it is important to define the notion of privacy. The framework of secure multiparty computation provides a solid theoretical underpinning for privacy. The key notion is to show that a protocol reveals nothing except the results. This is done by showing how everything seen during the protocol can be simulated from knowing the input and the output of the protocol. Yao first postulated the two-party comparison problem (Yao's Millionaire Protocol) and developed a provably secure solution. This was extended to multiparty computations (for any computable functionality) by Goldreich et al. [20] and to the malicious model of computation by Ben-Or et al. [11]. Overall, a framework was developed for secure multiparty computation. Goldreich [19] shows that computing a function privately is equivalent to computing it securely. We now cover some of the different models of computation in SMC.

9.1 Trusted Third Party Model

The gold standard for security is the assumption that we have a trusted third party to whom we can give all data. The third party performs the computation and delivers only the results {except for the third party, it is clear that nobody learns anything not inferable from its own input and the results. The goal of secure protocols is to reach this same level of privacy preservation, without the (potentially insoluble) problem of finding a third party that everyone trusts.

9.2 Semi-honest Model

The Semi-honest model is also known in the literature as the honest-but-curious model. A semi-honest party follows the rules of the protocol using its correct input, but after the protocol is free to use whatever it sees during execution of the protocol to compromise security / privacy.

Conclusion

In this paper we analyzed the different distributed database approach for dividing the centralized database into distributed database by using partitioning algorithm, and also this paper gives the details to how to preserve our distributed database from unauthorized access from the unauthenticated persons.

References

[1] Agrawal S, Krishnan V & Haritsa J. R. (2004). On addressing efficiency concerns in privacy-preserving mining. Proc. of the 9th Int'l Conf. on Database Systems for Advanced Applications. LNCS 2973, pp.113-124, Jeju Island: Springer-Verlag.

[2] Peng Z., Yun-Hai T., Shi-Wei T., Dong-Qing Y., & Xiu-Li M.. (August 2006). an Effective Method for Privacy Preserving Association Rule Mining. Journal of Software, 17(8), pp. 1764-1773.

[3] Yao A.C., (1982). Protocol for secure computations, In proceedings of the 23rd annual IEEE symposium on foundation of computer science, pp. 160-164.

[4] Gold reich O., Micali S. & Wigderson A.,(1987). How to play any mental game, in proceedings of the 19th annual ACM Symposium on Theory of Computation, pp. 218-229.

[5]Linedell Y. & Pinkas B., (2009) Secure Multiparty Computation for Privacy-Preserving Data Mining, The Journal of Privacy and Confidentiality, pp. 59-98.

[6] Paillier P., (1999). Public-key cryptosystems based on composite degree residuosity classes, Advances in Cryptography - EUROCRYPT '99, pp. 223-238, Prague, Czech Republic.

[7]Rivest R., Adleman L., & Dertouzos M..(1978). on data banks and privacy homomorphisms. In Foundations of Secure Computation, eds. R. A. De Milloetal., Academic Press, pp. 169-179.

[8]Qiong G., & Xiao-hui C. (2009). A privacy preserving distributed for mining association rules. International Conference on Artificial Intelligence and Computational Intelligence, pp. 294-297.

[9]. Koblitz N., (1994). A Course in Number Theory and Cryptography, Springer, Second edition.

[10]Stallings W., Cryptography and Network Security, Pearson Education, Third Edition.

[11]Apostol T., (1989) Introduction to Analytical Number Theory, Springer International, Student edition.

[12]Boneh D. & Shacham H., (2002).Fast Variants of RSA. Crypto Bytes, Springer, 5 (1) pp.1-9.

[13]Sun H. M. & Wu M. E., (2005). An Approach towards Rebalanced RSA-CRT with Short Public Exponent. Cryptology ePrint Archive: Report 2005/053.

[14]Ordonez C. & Chen Z., (2012). Horizontal Aggregations in SQL to Prepare Data Sets for Data Mining Analysis", IEEE Transactions on Knowledge and Data Engineering.

[15]Zhu J. (2009). A New Scheme to Privacy-Preserving Collaborative Data Mining, Information Assurance and Security IAS '09 Fifth International Conference, pp.18-20.

[16] Kumbhar, M.N., & Kharat, R. (2012). Privacy preserving mining of Association Rules on horizontally and vertically partitioned data: A review, Hybrid Intelligent Systems (HIS), 12th International Conference, pp. 4-7.

[17] Miyaji A.,& Rahman M.S. (2012). Privacy-Preserving Set Operations in the Presence of Rational Parties, Advanced Information Networking and Applications Workshops (WAINA), 26th International Conference, pp. 26-29.

[18] Raju R., Komalavalli R., & Kesavakumar V. (2009). Privacy Maintenance Collaborative Data Mining - A Practical Approach, 2nd International Conference on, Emerging Trends in Engineering and Technology (ICETET), pp. 16-18.

[19] Saleh I., Mokhtar A., Shoukry A., & Eltoweissy M. (2006). P3ARM: Privacy-Preserving Protocol for Association Rule Mining", Information Assurance Workshop IEEE.

[20]Ouyang W., & Huang Q., (2006). Privacy Preserving Sequential Pattern Mining Based on Secure Two-Party Computation", Information Acquisition, IEEE International Conference.