ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

Privacy Preserving Data Mining

Shaikh Imtiyaj

C V Raman College of Engineering, BPUT, Bhubaneswar, Odisha, India

ABSTRACT: With the rapid growth of the Internet, there is much need to cooperate mining data on the joint databases of multi-participants. Data mining is the extraction of hidden predictive information from large databases and also a powerful new technology with great potential to analyze important information in their data warehouses. Privacy preserving data mining is a latest research area in the field of data mining which generally deals with the side effects of the data mining techniques. Privacy is defined as "protecting individual's information". Protection of privacy has become an important issue in data mining research. Sensitive outlier protection is novel research in the data mining research field. Privacy Preserving Data mining techniques depends on privacy, which captures what information is sensitive in the original data and should therefore be protected from either direct or indirect disclosure. Secrecy and anonymity are useful ways of thinking about privacy. This privacy should be measureable and entity to be considered private should be valuable. In this paper, we discuss the various techniques that can be used for privatizing data. The goal is to secure access to confidential information while at the same time releasing aggregate information to the public. The challenge in each of the techniques is to protect data so that they can be published without revealing confidential information that can be linked to specific individuals. Also protection is to be achieved with minimum loss of the accuracy sought by database users. Different data partitioned have been discussed and a comparison of the same has been provided.

KEYWORDS: Data mining, Privacy, Data partitioning, Horizontal partitioning, Vertical Partitioning

I. INTRODUCTION

In the IT World, the networking and databases technologies enable data to be distributed across multi parties and gathered for sharing information. With the rapid growth of the Internet, there is much need to cooperate mining data on the joint databases of multi-participants. Data mining is the extraction of hidden predictive information from large databases and also a powerful new technology with great potential to analyze important information in their data warehouses. Privacy preserving data mining is a latest research area in the field of data mining which generally deals with the side effects of the data mining techniques. Privacy is defined as "protecting individual's information". Protection of privacy has become an important issue in data mining research. Sensitive protection is novel research in the data mining research field.

Privacy Preserving Data mining techniques depends on privacy, which captures what information is sensitive in the original data and should therefore be protected from either direct or indirect disclosure. Secrecy and anonymity are useful ways of thinking about privacy. This privacy should be measureable and entity to be considered private should be valuable.Distributed data mining such as association rule mining and decision tree learning are widely used by global enterprises. Data mining generally assumes a centralized server that collects data from multiple parties before performing data mining on the server. It generally assumes that data on the server can be shared among several parties. Privacy-preserving data mining (PPDM) was introduced to enable conventional data mining techniques to preserve data privacy during the mining process.

Privacy preserving data mining is a new research direction in data mining and knowledge discovery. The main reason for the rapid development of this research area is the growing awareness of the accumulation of huge amounts of easily available data on the Internet – data that may involve a threat to the privacy of users.

Privacy Preserving is the relationship between collection and dissemination of data, technology, the public expectation of privacy, and the legal and political issues surrounding them. Privacy concerns exist wherever personally identifiable information is collected and stored in digital form or otherwise. Improper or non-existent disclosure control can be the root cause for privacy issues. Privacy issues can arise in response to information from a wide range of sources, such as: Healthcare records, Criminal justice investigations and proceedings, financial institutions and

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

transactions, Biological traits, such as genetic material, Residence and geographic records .The challenge in privacy preserving is to share data while protecting personally identifiable information. The fields of data security and information security design and utilize software, hardware and human resources to address this issue. The main goal of Privacy preserving data mining is to mine the rules or pattern accurately without revealing any other private information.

A. DATA MINING

Data mining (knowledge discovery from data) is the extraction of interesting (non-trivial, implicit, previously unknown and potentially useful) patterns or knowledge from huge amount of data.
Data mining is also known as Knowledge discovery (mining) in databases (KDD), knowledge extraction, data/pattern analysis, data archaeology, data dredging, information harvesting, business intelligence, etc.

Knowledge Discovery (KDD) Process

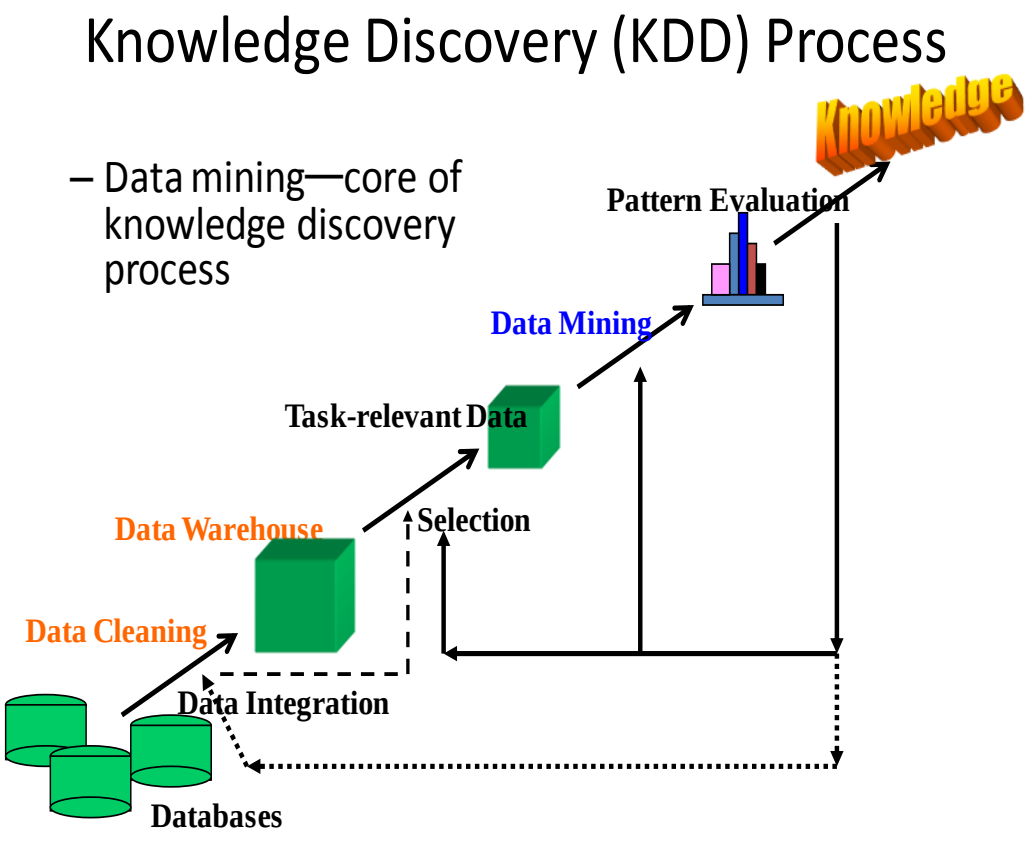

Figure 1.1: KDD Process

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

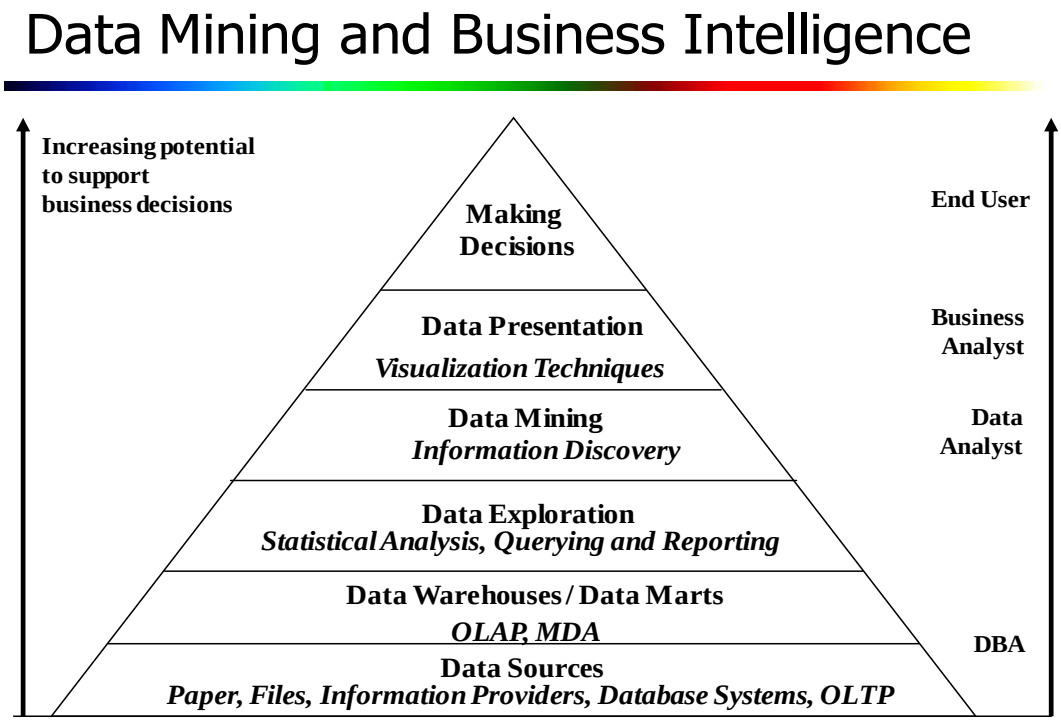

Figure 1.2: DM and Business Intelligence

II. LITERATURE SURVEY

A survey on Privacy-preserving data mining finds numerous applications in surveillance which are naturally supposed to be "privacy-violating" applications. The key is to design methods [1,2] which continue to be effective, without compromising security, a number of techniques have been discussed for bio-surveillance, facial de-dentification, and identity theft. More detailed discussions on some of these issues may be found in [3,4]. Most methods for privacy computations use some form of transformation on the data in order to perform the privacy preservation. Typically, such methods reduce the granularity of representation in order to reduce the privacy. This reduction in granularity results in some loss of effectiveness of data management or mining algorithms. This is the natural trade-off between information loss and privacy. Data mining generally assumes data on the server can be shared among several parties and as privacy issues become more prevalent. Privacy-preserving data mining was introduced [5], [6][7][8] to enable conventional data mining techniques to preserve data privacy during the mining process. Some work has been done to explore privacy-preserving data mining on horizontally and/or vertically partitioned data involving multiple parties so that no single party holds the overall data [9][10][11]. In horizontally partitioned data two or more parties hold different objects for the same set of attributes. It means each object in the virtual database is completely owned by one party. For vertically partitioned data, two parties or more hold the different set of attributes for the same set of objects. In arbitrarily partitioned data, different disjoint portions are held by different parties. This is perhaps the most general form of data partitioning, as introduced by Jagannathan and Wright [12] for two parties. As argued by the authors, although extremely "patch worked" data is unlikely in practice, it is better suited to practical settings as a more general model of horizontally and vertically partitioned data. The secure scalar product is a core operation in decision tree induction for vertically partitioned data. Much work has been done that discussed how the secure scalar product can be computed for two parties .Vaidya and Clifton introduced the Secure Set Intersection Cardinality method to perform secure scalar product for multiple parties [13]. This method has been applied to perform decision tree induction [14] and association rule mining for vertically partitioned data, and SVM model construction for horizontally partitioned data. A major weakness of the Secure Set Intersection Cardinality is its computational and communication complexities, which are $O(mn)$ and $O(mn2)$ respectively, where n is the number of parties and m is the length of private vectors. In

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

arbitrary partitioned the decision tree induction can be performed data partition involving two parties, which is similar to the case for vertically partitioned data. Then, extended to *n* parties and propose a protocol to securely compute PSP with computational and communication complexities of $O(n)$ and $O(mn)$ respectively.

In this thesis the different techniques have been used for privacy preserving. We focus on classification problem, and present a novel privacy preserving decision tree method. Theoretical analysis and experimental results show that it gives good capability of privacy preserving, efficiency and accuracy.

A. Multi party Privacy Preserving Decision trees for Vertically partitioned Data

For the vertically partitioned case, many primitive operations such as computing the scalar product or the secure set size intersection can be useful in computing the results of data mining algorithms. For example, the methods discuss how to use to scalar dot product computation for frequent item set counting. The process of counting can also be achieved by using the secure size of set intersection as described in [15]. Another method for association rule mining discussed uses the secure scalar product over the vertical bit representation of item set inclusion in transactions, in order to compute the frequency of the corresponding item sets. This key step is applied repeatedly within the framework of a roll up procedure of item set counting. It has been shown that this approach is quite effective in practice. The approach of vertically partitioned mining has been extended to a variety of data mining applications such as decision trees [16], SVM Classification [17],Naïve Bayes Classifier[18] and k-means clustering [19]. A number of theoretical results on the ability to learn different kinds of functions in vertically partitioned databases with the use of cryptographic approaches are discussed in [20].

B. Multi party Privacy Preserving Decision trees for horizontally partitioned Data

In horizontally partitioned data sets, different sites contain different sets of records with the same (or highly overlapping) set of attributes which are used for mining purposes. Many of these techniques use specialized versions of the general methods discussed in [21, 22] for various problems. The work in [23] discusses the construction of a popular decision tree induction method called ID3 with the use of approximations of the best splitting attributes. Subsequently, a variety of classifiers have been generalized to the problem of horizontally partitioned privacy preserving mining including the Naïve Bayes Classifier [24], and the SVM Classifier with nonlinear kernels [25]. An extreme solution for the horizontally partitioned case is discussed in [26], in which privacy preserving classification is performed in a fully distributed setting, where each customer has private access to only their own record. A host of other data mining applications have been generalized to the problem of horizontally partitioned data sets. These include the applications of association rule mining, clustering and collaborative filtering. A related problem is that of information retrieval and document indexing in a network of content providers. This problem arises in the context of multiple providers which may need to cooperate with one another in sharing their content, but may essentially be business competitors. it has been discussed how an adversary may use the output of search engines and content providers in order to reconstruct the documents. Therefore, the level of trust required grows with the number of content providers. A solution to this problem constructs a centralized privacy-preserving index in conjunction with a distributed access control mechanism. The privacy-preserving index maintains strong privacy guarantees even in the face of colluding adversaries, and even if the entire index is made public.

C. The k-anonymity Method

In many cases, it is important to maintain k-anonymity across different distributed parties. In [27] a k-anonymous protocol for data which is vertically partitioned across two parties is described. The broad idea is for the two parties to agree on the quasi-identifier to generalize to the same value before release. The work in [28] discusses an extreme case in which each site is a customer which owns exactly one tuple from the data. It is assumed that the data record has both sensitive attributes and quasi-identifier attributes. The solution uses encryption on the sensitive attributes. The sensitive values can be decrypted only if therefore are at least k records with the same values on the quasi-identifiers. Thus, k-anonymity is maintained.

The issue of k-anonymity is also important in the context of hiding identification in the context of distributed location based services [29]. In this case, k-anonymity of the user-identity is maintained even when the location information is released. Such location information is often released when a user may send a message at any point from a given location. A similar issue arises in the context of communication protocols in which the anonymity of senders (or

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

receivers) may need to be protected. A message is said to be *sender* k-*anonymous*, if it is guaranteed that an attacker can at most narrow down the identity of the sender to k individuals. Similarly, a message is said to be *receiver* k-*anonymous*, if it is guaranteed that an attacker can at most narrow down the identity of the receiver to k individuals. A number of such techniques have been discussed.

III. PRIVACY PRESERVING DATA MINING MODELS AND METHODS

A. ARCHITECTURE OF DATA MINING SYSTEM

The field of privacy has seen rapid advances in recent years because of the increases in the ability to store data. In particular, recent advances in the data mining field have lead to increased concerns about privacy. The problem of privacy-preserving data mining has become more important in recent years because of the increasing ability to store personal data about users, and the increasing sophistication of data mining algorithms to leverage this information. The architecture of data mining is shown below:

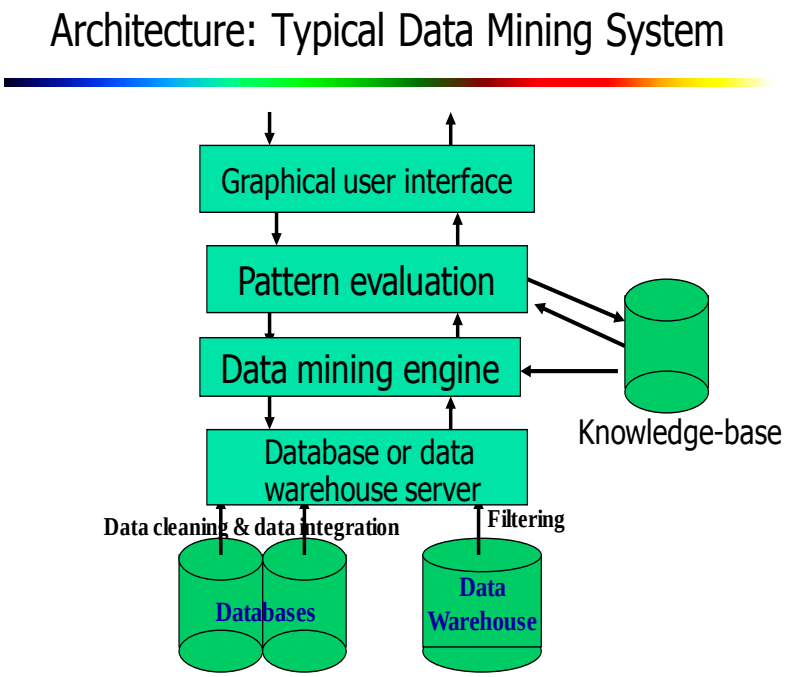

Figure 2.1: Architecture of Data Mining

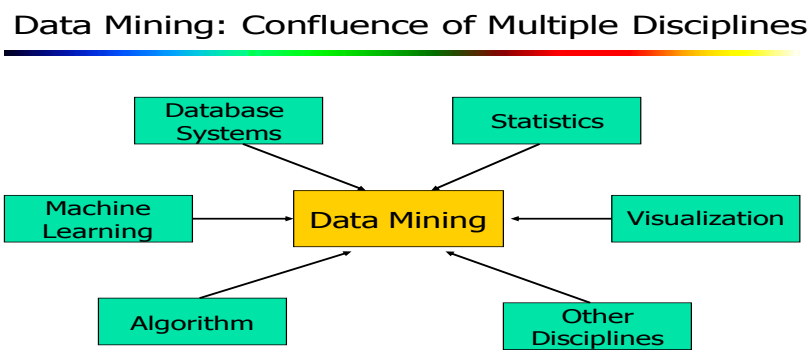

Figure 2.2: Data Mining Confluence of Multiple Disciplines

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

B. Cryptographic Methods for Information Sharing and Privacy

In many cases, multiple parties may wish to share *aggregate private data*, without leaking any sensitive information at their end. For example, different superstores with sensitive sales data may wish to coordinate among themselves in knowing aggregate trends without leaking the trends of their individual stores. This requires *secure and cryptographic protocols* for sharing the information across the different parties. The data may be distributed in two ways across different sites:

Horizontal Partitioning: the different sites may have different sets of records containing the same attributes.

Vertical Partitioning: In this case, the different sites may have different attributes of the same sets of records

IV. EXPERIMENTAL RESULTS

A. MULTI PARTY PRIVACY PRESERVING DECISION TREE FOR VERTICALLY PARTITIONED DATA

Decision Tree

The Decision Tree is one of the most popular classification algorithms in current use in Data Mining and Machine Learning. In this paper, we propose multi party privacy preserving distributed decision tree method based on ID3 , which is applied in mining concentrative database and uses information entropy to choose the best prediction attribute.

A decision tree consists of nodes and arcs which connect nodes. To make a decision, one starts at the root node, and asks questions to determine which arc to follow, until one reaches a leaf node and the decision is made. This basic structure is shown in Figure below.

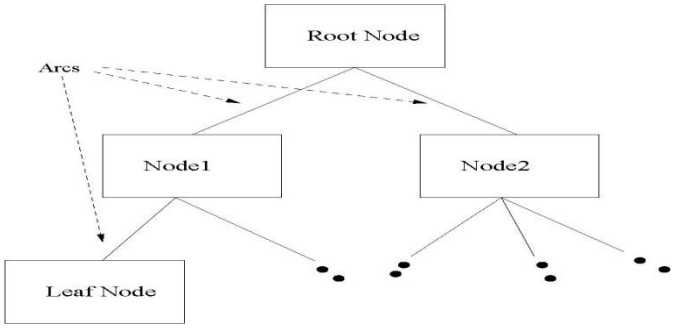

Figure 3.1: Basic Decision tree structure

Partitioned Data

Partitioning in general, results in smaller, manageable data sizes, so indexes are built faster, queries run faster; more data can actually fit into memory, and so on. Partitioning a database improves performance and simplifies maintenance. By splitting a large table into smaller, individual tables, queries that access only a fraction of the data can run faster because there is less data to scan. With Vertical Partitioning, we break a large table, typically with lot of columns, into multiple tables based on certain criteria so as to minimize the need for querying across those multiple tables. So, after the vertical partitioning into 3, a large table with 'n' rows and 'm' columns may look like:
table_partition_1: 'n' rows, 'x' columns
table_partition_2: 'n' rows, 'y' columns
table_partition_3: 'n' rows, 'z' columns
Where (x + y + z) = m
We consider the weather data details in which multi parties get there required data from Databases wish to run a data-mining algorithm on the union of their databases, without revealing any original information.

Tree building

The weather data set which we will use to understand how a decision tree is built.

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

Table 3.1: Detail information's of "Weather Data"

ID	Outlook	Temp	Humidity	Wind	Result
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no
3	overcast	hot	high	false	yes
4	rain	mild	high	false	yes
5	rain	cool	normal	false	yes
6	rain	cool	normal	true	no
7	overcast	cool	normal	true	yes
8	sunny	mild	high	false	no
9	sunny	cool	normal	false	yes
10	rain	mild	normal	false	yes
11	sunny	mild	normal	true	yes
12	overcast	mild	high	true	yes
13	overcast	hot	normal	false	yes
14	rain	mild	high	true	no

Let X is the set of condition attributes and C be the class attribute, we make assumptions that the database is vertically partitioned between n parties; each party Pi only knows its own attributes Xi, transaction ID and attribute C are known to all parties.
We take an example, as Table 3.1 shows; the class attribute is *Result*, which is determined by condition attributes, such as outlook, temp, humidity, wind, min_temp and max_temp.

Privacy-preserving algorithm

Assume that there are three parties named A, B and C, which respectively has *xa*, *xb* and *xc* condition attributes, and wants to collaboratively mining decision-tree.

Local mining algorithm (performed by parties with token):
Input: Local training samples, token.
Output: Sending class attribute distribution to miner site, or sending IDs to other parties and information entropy to miner.

1) If token=0, computes the class attribute value c assigned to most transactions with the certain IDs, and sends c to miner site;
2) If token=1, judges if all the transactions with the certain IDs have the same class attribute c, if so, sends c to miner site;

3) If not, works out the intersection of transactions used previously and transactions with best prediction attribute value, sends IDs to other parties by MNP(Mix Network Protocol), and do step 4;

4) Computes information entropy and sends it to the miner site by Protocol for Comparing Information Without Leaking (PCIWL);

Local mining algorithm (performed by parties without token):
Input: Local training samples, transaction IDs.
Output: Sending information entropy to miner.

1) Receives transaction IDs form the token party, then computes intersection of IDs received and IDs used previously;
2) Computes information entropy corresponding to the certain IDs, and sends it to the miner site by Protocol for Comparing Information Without Leaking (PCIWL).

Miner site algorithm (performed by miner):
Input: Class attribute distribution from token party, or information entropy from all parties.
Output: Creating node, updating Constraint Set, sending token signal to target party.

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

1) If the receiving message is class attribute c from token party, creates a leaf node with the value c;
2) If the receiving message is information entropy from all parties, applies Protocol for Comparing Information Without Leaking (PCIWL) to obtain maximum, and do step 3;
3) Creates an internal node with the value of target party's name and serial number of the best prediction attribute, adds the attribute to Constraint Set, and do step 4;
4) If Constraint Set is full, sends token=0 to the target party; otherwise sends token=1.

ID3 Algorithm

ID3(Examples, Target_attribute, Attributes)

1. Create a **Root** node for the tree
2. If all **Examples** are positive, Return the single-node tree **Root**, with label = +
3. If all **Examples** are negative, Return the single-node tree **Root**, with label = -
4. If **Attributes** is empty, Return the single-node tree **Root**, with label = most common value of **Target_attribute** in **Examples**
5. Otherwise Begin
   - **A** ← the attribute from **Attributes** that best classifies **Examples**
   - The decision attribute for **Root**← **A**
   - For each possible value, **v1**, of **A**,
     - Add a new tree branch below **Root**, corresponding to the test **A= v1**
     - Let **Examples** v1 be the subset of **Examples** that have value **v1** for **A**
     - If **Examples** v1 is empty
       - Then below this new branch add a leaf node with label = most common value of **Target_attribute** in **Examples**
       - Else below this new branch add the subtree

         ID3(**Examples**,**Target_attribute**,**Attributes** {A})
6. End
7. Return **Root**

Attribute Selection

How does ID3 decide which attribute is the best? A statistical property, called information gain, is used. Gain measures how well a given attribute separates training examples into targeted classes. The one with the highest information (information being the most useful for classification) is selected. In order to define gain, we first borrow an idea from information theory called entropy. Entropy measures the amount of information in an attribute.

Given a collection S of c outcomes

$$Entropy(S) = \sum -p(I) \log_2 p(I)$$

where p(I) is the proportion of S belonging to class I. S is over c. Log2 is log base 2.

Note that S is not an attribute but the entire sample set.

If S is a collection of 14 examples with 9 YES and 5 NO examples then

$$Entropy(S) = - (9/14) \text{ Log2 } (9/14) - (5/14) \text{ Log2 } (5/14) = 0.940$$

Notice entropy is 0 if all members of S belong to the same class (the data is perfectly classified). The range of entropy is 0 ("perfectly classified") to 1 ("totally random").

Gain(S, A) is information gain of example set S on attribute A is defined as

$$Gain(S, A) = Entropy(S) - \sum ((|S_v| / |S|) * Entropy(S_v))$$

Where: Σ is each value v of all possible values of attribute A, S_v = subset of S for which attribute A has value v, $|S_v|$ = number of elements in S_v, $|S|$ = number of elements in S

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

Here S is a set of 14 examples in which one of the attributes is wind. The values of Wind can be *false* or *true*. The classification of these 14 examples are 9 yes and 5 no. For attribute Wind, suppose there are 8 occurrences of Wind = false and 6 occurrences of Wind = true. For Wind = false, 6 of the examples are yes and 2 are no. For Wind = true, 3 are yes and 3 are no. Therefore

$Gain(S,Wind)=Entropy(S)-(8/14)*Entropy(S_{weak})-(6/14)*Entropy(S_{strong})$

$= 0.940 - (8/14)*0.811 - (6/14)*1.00$

$= 0.048$

$Entropy(S_{weak}) = - (6/8)*log2(6/8) - (2/8)*log2(2/8) = 0.811$

$Entropy(S_{strong}) = - (3/6)*log2(3/6) - (3/6)*log2(3/6) = 1.00$

For each attribute, the gain is calculated and the highest gain is used in the decision node.

We need to find which attribute will be the root node in our decision tree. The gain is calculated for all four attributes:

Gain(S, Outlook) = 0.246

Gain(S, Temperature) = 0.029

Gain(S, Humidity) = 0.151

Gain(S, Wind) = 0.048

Outlook attribute has the highest gain, therefore it is used as the decision attribute in the root node. Since Outlook has three possible values, the root node has three branches (sunny, overcast, rain). we only decide on the remaining three attributes: Humidity, Temperature, or Wind

S_{sunny} = {1, 2, 8, 9, 11} = 5 examples from table 1 with outlook = sunny

$Gain(S_{sunny}, Humidity) = 0.970$

$Gain(S_{sunny}, Temperature) = 0.570$

$Gain(S_{sunny}, Wind) = 0.019$

Humidity has the highest gain; therefore, it is used as the decision node. This process goes on until all data is classified perfectly or we run out of attributes. By using ID3 algorithm to mine on the union of datasets,

B. MULTI PARTY PRIVACY PRESERVING DECISION TREE FOR HORIZONTALLY PARTITIONED DATA

Partitioning in general, results in smaller, manageable data sizes, so indexes are built faster, queries run faster, more data can actually fit into memory, and so on. Partitioning a database improves performance and simplifies maintenance. By splitting a large table into smaller, individual tables, queries that access only a fraction of the data can run faster because there is less data to scan. Maintenance tasks, such as rebuilding indexes or backing up a table, can run more quickly. Partitioning can be achieved without splitting tables by physically putting tables on individual disk drives. Putting a table on one physical drive and related tables on a separate drive can improve query performance because, when queries that involve joins between the tables are run, multiple disk heads read data at the same time.

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

With Horizontal Partitioning, we keep the columns in a large table intact, but split the rows, again based on certain criteria so as to minimize querying across multiple partitions. a horizontally partitioned table might look like:
table_partition_1: n/k rows, 'm' columns
table_partition_2: n/k rows, 'm' columns
…..
table_partition_k: n/k rows, 'm' columns

Horizontally partitioned

1. Partition by time into equal segments
2. Partition by time into different sized segments
3. Partition on a different dimension, e.g. region
4. Partition by size of table

Table 4.1(a): User_1

ID	Outlook	Temp	Humidity	Wind	Result
1	sunny	hot	high	false	no
2	sunny	hot	high	true	no

:
:

Table 4.1(f): User_7

ID	Outlook	Temp	Humidity	Wind	Result
13	overcast	hot	normal	false	yes
14	rain	mild	high	true	no

Table 4.1 Accessed Data by User

By using ID3 algorithm to mine on the union of datasets, we can obtain the public decision tree, while each party's private information are all revealed. We study and analysis the multi party privacy preserving decision trees for horizontally partitioned data that when building decision tree, the control is passing from site to site, except token party has the knowledge of best prediction attribute of the present node, other party even the miner doesn't know any relevant information. Entropy is calculated as

Table 4.2: Entropy Estimation

Entropy	value
Entropy(S)	0.940
Entropy(S_{weak})	0.811
Entropy(S_{strong})	1.00

Gain is calculated as

Table 4.3: Gain Estimation

Gain(S, Outlook)	0.246
Gain(S, Temperature)	0.029
Gain(S, Humidity)	0.151
Gain(S, Wind)	0.048

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

Gain(S_{sunny}, Humidity)	0.970
Gain(S_{sunny}, Temperature)	0.570
Gain(S_{sunny}, Wind)	0.019

V. CONCLUSION

In this thesis work, we presented a survey of the broad areas of privacy-preserving data mining and the underlying algorithms. We discussed a variety of data modification techniques such as randomization and k-anonymity based techniques and finally partitioning method. We discussed methods for distributed privacy-preserving mining, and the methods for handling horizontally and vertically partitioned data. We discussed the issue of downgrading the effectiveness of data mining and data management applications such as association rule mining, classification, and query processing. We also discussed some fundamental limitations of the problem of privacy preservation in the presence of increased amounts of public information and background knowledge. Finally, a number of diverse application domains for which privacy-preserving data mining methods are useful have been narrated. The primary contribution of this work is to propose a multi party privacy preserving decision tree for vertically partitioned data and also horizontally partitioned data by using ID3 algorithm. This has particular relevance to privacy-sensitive searches, particularly top-k queries, and meshes well with privacy policies such as k-anonymity. There remain many open problems in developing secure solutions based on efficient non secure query processing algorithms. Further, this work has shown that there is a trade-off between efficiency and the amount of information that is disclosed. It is worthwhile to explore whether one could have a suite of algorithms (or a configurable algorithm) so that applications can choose the goal they want to optimize.

We study and analysis the multi party privacy preserving decision trees for vertically partitioned data that when building decision tree, the control is passing from site to site, except token party has the knowledge of best prediction attribute of the present node, other party even the miner doesn't know any relevant information. When classifying, the miner only knows the path of classifying process, i.e., which site handles the classifying in every step, while the information of which attribute is used to classify and values of transaction records in every party is protected. Finally this gives the best privacy preserving, efficiency and accuracy.

REFERENCES

[1] Grljevic O,Bosnjak Z,Mekovrc R., "Privacy Preserving in Data Mining- Experimental research on SMEs data", *2011 IEEE International Symposium on Privacy Preserving*, Vol.4, pp. 477-481, October 2011

[2] K. Murat and Chris 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, September 2004

[3] Newton E., Sweeney L., Malin B., "Preserving Privacy by De-identifying Facial Images", *IEEE Transactions on Knowledge and Data Engineering, IEEE TKDE*, pp.1013-1021, February 2005.

[4] Li, Y., Chen, M., Li, Q. Zhang, W., " Enabling Multi-Level Trust in Privacy Preserving Data Mining ", *IEEE Transactions on Knowledge and Data Engineering* ,Volume**:** PP Issue:99, page 1-11,09 ,June 2011

[5] [5] Zhe Jia, Lei Pang, Shoushan Luo, Yang Xin.: Miao Zhang, "Research on Distributed Privacy- Preserving Data Mining", *JCIT: Journal of Convergence Information Technology*, Vol. 7, No. 1, pp. 356-367, 2012

[6] [6] Y. Lindell and B. Pinkas., "Privacy preserving data mining", In *Advances in Cryptology*, *volume 1880 of Lecture Notes in Computer Science*, pp. 36–53, Springer-Verlag, 2000

[7] [7] J vaidya, C Clifton., "Privacy Preserving Kth Element Score over Vertically Partitioned Data", *IEEE Transaction on Knowledge and Data Engineering*, Vol.21,No.2, pp.253-258, February 2009

[8] [8] Sumana M and Dr Hareesh K S., "Anonymity: An Assessment and Perspective in Privacy Preserving Data Mining", *International Journal of Computer Applications* Record 6(10), pp.1–5, September 2010.

[9] [9] V. S. Verykios, E. Bertino, I. N. Fovino, L. P. Provenza, Y. Saygin, and Y. Theodoridis., "State-of-the-art in privacy preserving data mining", *ACM SIGMOD Record*, 3(1), pp. 50–57, March 2004.

[10] [10] Nan Zhang ,Wei Zhao, "Privacy Protection against Malicious Adversaries in Distributed Information Sharing Systems", *IEEE Transaction on Knowledge and Data Engineering*, Vol.2.0, No.8, pp. 1028-1033, August 2008

[11] [11] H. Yu, X. Jiang, and J. Vaidya.: "Privacy-preserving svm using nonlinear kernels on horizontally partitioned data", *In Proceedings of the ACM Symposium on Applied Computing*, pp. 603–610, Dijon, France, 2006.

[12] [12] Bawa M., Bayardo R. J., Agrawal and J Vaidya, "Privacy-Preserving Indexing of Documents on the Network", *VLDB Conference*, pp.23-30, 2003.

[13] J. Vaidya and C. Clifton.: "Secure set intersection cardinality with application to association rule mining", *Journal of Computer Security*, 13(4), pp. 24-31, July 2005.

[14] Jaideep Vaidya ,Chris Clifton, "Privacy-Preserving decision trees over Vertically Partitioned Data, Proceedings of the 19th annual IFIP working conference on Data and Applications Security, pp.139-152, August 2005,

ISSN: 2350-0328

International Journal of Advanced Research in Science, Engineering and Technology

Vol. 2, Issue 2 , February 2015

[15] Jaideep Vaidya , Chris Clifton, "Privacy preserving association rule mining in vertically partitioned data", *Proceedings of the eighth ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 114-120, 2002, Edmonton, Alberta, Canada
[16] Jaideep Vaidya , Chris Clifton , Murat Kantarcioglu , A. Scott Patterson, "Privacy-Preserving Decision trees over Vertically Partitioned Data", ACM Transactions on Knowledge Discovery from Data (TKDD), Vol.2, No.3, pp.1-27, October 2008
[17] Hwanjo Yu , Jaideep Vaidya , Xiaoqian Jiang, "Privacy-Preserving SVM classification on Vertically Partitioned Data", Proceedings of the 10th Pacific-Asia conference on Advances in Knowledge Discovery and Data Mining, pp. 540-546, April 2006,
[18] Jaideep Vaidya , Murat Kantarcıoğlu , Chris Clifton, "Privacy Preserving Naive Bayes Classification", The VLDB Journal — The International Journal on Very Large Data Bases, Vol.17 ,No.4, pp. 879-898, July 008
[19] Vaidya J., Clifton C.: "Privacy-Preserving K-means clustering over Vertically Partitioned Data", *ACM KDD Conference*, pp.60-65, 2003.
[20] Wang K., Fung B. C. M., Dong G.: "Integarting Private Databases for Data Analysis", *Lecture Notes in Computer Science*, 3495, pp.1-24, 2005.
[21] Clifton C.,Kantarcioglou M., LinX., ZhuM.: "Tools for privacy-preserving distributed data mining", *ACM SIGKDD Explorations*, 4(2),pp. 342-348, 2002.
[22] Du W., Atallah M.: "Secure Multi-party Computation, A Review and Open Problems", *CERIAS Tech. Report* 2001-51, pp.1-51, Purdue University, 2001.
[23] Lindell Y., Pinkas B.: "Privacy-Preserving Data Mining", *CRYPTO*, pp.176-190, 2000.
[24] Kantarcioglu M., Vaidya J.: "Privacy-Preserving Naive Bayes Classifier for Horizontally Partitioned Data" *IEEE Workshop on Privacy-Preserving Data Mining*, pp.256-261, 2003
[25] Yu H., Jiang X., Vaidya J.: "Privacy-Preserving SVM using nonlinear Kernels on Horizontally Partitioned Data", *SAC Conference*, pp.312-317, 2006
[26] Murat Kantarcioglu , Chris 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, September 2004
[27] P. Samarati and L. Sweeney, "Protecting privacy when disclosing information: *k*-anonymity and its enforcement through generalization and suppression", *Proceedings of the IEEE Symposium on Research in Security and Privacy*, pp.670-678, Oakland, CA, 1998.
[28] Zhong S., Yang Z., Wright R.: "Privacy-enhancing K-anonymization of customer data", *Proceedings of the ACM SIGMOD-SIGACT-SIGART Principles of Database Systems*, pp.173-182, Baltimore, MD. 2005.
[29] Bettini C., Wang X. S., Jajodia S., "Protecting Privacy against Location Based Personal Identification", *Proc. of Secure Data Management Workshop*, pp. 185-199, Trondheim, Norway, 2005.