International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

Data mining using RDT on Encryption & Decryption data Cryptography technique

B.Madhusudhana Reddy[1]; S.G. Nawaz[2]& M.Harathi[3]
M-Tech Dept of Computer science Sri Krishna Devaraya Engineering college, Gooty, Anantapur Email: - madhu1214.b@gmail.com
ASSOC PROFESOR Dept of Computer science Sri Krishna Devaraya Engineering college, Gooty, Anantapur Email: - harathi8@gmail.com
HOD Dept of Computer science Sri Krishna Devaraya Engineering college, Gooty, Anantapur Email: - sngnawaz@gmail.com

Abstract
Privacy preservation in data mining has been a popular and an important research area for more than a decade due to its vast spectrum of applications. A new class of data mining method called privacy preserving data mining algorithm has been developed. The aim of this algorithm is to protect the sensitive information in data from the large amount of data set. The privacy preservation of data set can be expressed in the form of decision tree, cluster or association rule. This paper proposes a privacy preservation based on data set complement algorithms which store the information of the real dataset. So that the private data can be safe from the unauthorized party, if some portion of the data can be lost, then we can reconstructed the original data set from the unrealized dataset and the perturbing data set.

Keyword: Privacy preserving RDT; Data mining; Encryption; Decryption; Cryptography technique.

1. Introduction

Data Mining is quick growing field of distributed atmosphere and method of discovering fascinating patterns and knowledge from giant information. It's additionally known as KDD process i.e. information Discovery from knowledge. It permits knowledge analysis whereas conserving knowledge privacy. Data privacy conserving is forestalling personal secret or non-public data from unnecessarily distributed or in public identified or not be put-upon by person or by oppose. In privacy preserving data processing, fascinating and helpful data is distributed with privacy of guidance has been preserved. There square measure 2 stages in privacy conserving knowledge mining initial is knowledge assortment and second knowledge commercial enterprise. In data assortment, knowledge holder stores knowledge that is gathered by data owner. In knowledge commercial enterprise, knowledge may be free to knowledge recipient by knowledge holder and knowledge recipient mines printed secured knowledge. Cryptographic techniques square measure typically too slow to be sensible and can become computationally expensive because the rise in size of the info set and communications between numerous parties increase [1]. Crypto graphical techniques cannot handle huge data. During this paper, we tend to square measure victimization privacy conserving RDT is Random decision Tree with privacy conserving data processing which is developed by Fan et al. [3]. Privacy conserving RDT is combination of randomization and cryptography technique.

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

This resolution provides Associate in nursing order of magnitude improvement inefficiency over existing solutions whereas providing a lot of knowledge privacy and knowledge utility. This can be an efficient resolution to privacy-preserving data processing for the massive knowledge challenge. Random decision Tree provides higher potency and knowledge privacy than crypto graphical technique. RDT provides a structural property, a lot of specifically, the very fact that solely specific nodes (the leaves) within the classification tree have to be compelled to be encrypted /decrypted, and secure token passing prevents adversary from utilizing count techniques to decipher instance classifications, because the branch structure of the tree is hidden from all parties. RDT to get trees. That square measure random in structure, providing USA with an analogous finish result as perturbation while not the associated pitfalls. A random structure provides security against investing priority information to get the whole classification model or instances.

2. Related Work

A decision tree[3][4][5] is defined as "a predictive modeling technique from the field of machine learning and statistics that builds a simple tree-like structure to model the underlying pattern of data". Decision tree is one of the popular methods is able to handle both categorical and numerical data and perform classification with less computation. Decision trees are often easier to interpret. Decision tree is a classifier which is a directed tree with a node having no incoming edges called root. All the nodes except root have exactly one incoming edge. Each non-leaf node called internal node or splitting node contains a decision and most appropriate target value assigned to one class is represented by leaf node. Decision tree classifier is able to break down a complex decision making process into collection of simpler decision. The complex decision is subdivided into simpler decision on the basis of splitting criteria. It divides whole training set into smaller subsets. Information gain, gain ratio, gini index are three basic splitting criteria to select attribute as a splitting point. Decision trees can be built from historical data they are often used for explanatory analysis as well as a form of supervision learning. The algorithm is designed in such a way that it works on all the data that is available and as perfect as possible. According to Breiman *et al.* [6] the tree complexity has a crucial effect on its accuracy performance. The tree complexity is explicitly controlled by the pruning method employed and the stopping criteria used. Usually, the *tree complexity* is measured by one of the following metrics:

- The total number of nodes;
- Total number of leaves;
- Tree depth;
- Number of attributes used.

Decision tree induction is closely related to rule induction. Each path from the root of a decision tree to one of its leaves can be transformed into a rule simply by conjoining the tests along the path to form the antecedent part, and taking the leaf's class prediction as the class value. The resulting rule set can then be simplified to improve its accuracy and comprehensibility to a human user [7].

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

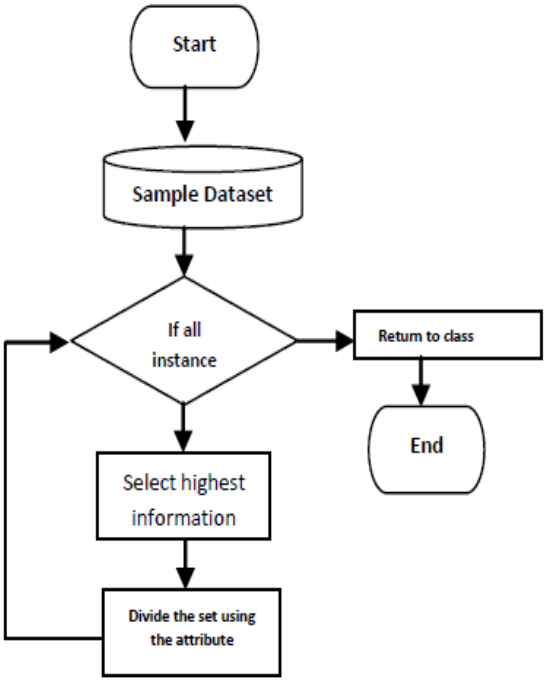

Fig 1: Flowchart for tree based classification

Hyafil and Rivest proved that getting the optimal tree is NP-complete [8]. Most algorithms employ the greedy search and the divide-and-conquer approach to grow a tree. In particular, the training data set continues to be split in small. The related algorithm ID3 and C4.5 [9] adopt a greedy approach in which decision trees are constructed in top down recursive divide and conquer manner. ID3 was one of the first Decision tree algorithms. It works on wide variety of problems in both academia and industry and has been modified improved and borrowed from many times over. ID3 picks splitting value and predicators on the basis of gain in information that the split or splits provide. Gain represents difference between the amount of information that is needed to correctly make a prediction both before and after the split has been made. Information gain is defined as the difference between the entropy of original segment and accumulated entropies of the resulting split segment. C4.5 is an extension of ID3, presented by the same author (Quinlan, 1993). It uses gain ratio as splitting criteria.

The splitting ceases when the number of instances to be split is below a certain threshold. C4.5 can handle numeric attributes. It performs error based pruning after growing phase. It can use corrected gain ratio induce from a training set that incorporates missing values.

Example:

Table 1 shows the well known weather data set distributed between two parties. We assume that when the data set is horizontally partitioned, instances 1-7 are owned by Party 1,

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

	—P1—		—P2—		
	outlook	temperature	humidity	windy	play
P1	sunny	hot	high	weak	no
	sunny	hot	high	strong	no
	overcast	hot	high	weak	yes
	rainy	mild	high	weak	yes
	rainy	cool	normal	weak	yes
	rainy	cool	normal	strong	no
	overcast	cool	normal	strong	yes
P2	sunny	mild	high	weak	no
	sunny	cool	normal	weak	yes
	rainy	mild	normal	weak	yes
	sunny	mild	normal	strong	yes
	overcast	mild	high	strong	yes
	overcast	hot	normal	weak	yes
	rainy	mild	high	strong	no

Table 1: The Distributed Weather Data Set.

While 8-14 are owned by Party 2.If it is vertically partitioned, we assume that Party 1 owns the outlook and temperature attributes while Party 2 owns the humidity, windy, and play attributes. To save space, both cases are shown together in Table 1. For simplicity, we assume that only two RDT trees are built (depicted in Fig. 1). Suppose a new instance {sunny, mild, normal, weak} is to be classified. Then, as per the first random tree, the prediction is (2, 0) without normalization. The prediction as per the second random tree is (1, 2). Therefore, the non-normalized overall class distribution vector provided by RDT is (1.5, 1).

3. Implementation

3.1 DATA SET COMPLEMENTATION APPROACH:

Privacy preservation via dataset complementation is a data perturbed approach that substitutes each original dataset with an entire unreal dataset. Unlike privacy protection strategies, this new approach preserves the original accuracy of the training datasets without linking the perturbed datasets to the information providers. In other words, dataset complementation can preserve the privacy of individual records and yield accurate data mining results. However, this approach is designed for discrete-value classification only, such that ranged values must be defined for continuous values.

A. Universal Set and Data Set Complement:

In set theory, a universal set U is a set which contains all elements [20]. In this paper, a universal set U^T, relating to a data table T, is a set of datasets that contains a single instance of each valid dataset of T. In other words, any combination of a possible value from each attribute in the dataset sequence of T exists in U^T. If t is a dataset in T associated with a tuple of attributes<a1, a2, , , am>

International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

a_i has n_i possible values $K_i = \{k_1, k_2 \ldots k_{mi}\}$, then $< I[a_1], I[a_2], \ldots I[a_i] \ldots, I[a_m] > \in T^U$ and $I[a_i] \in K_i$.

We define:

U T is a set containing a single instance of all possible datasets in data tableT . The table associates with attributes < Outlook,Humidity,Wind,Play >and possible attribute values are defined as: Weather = {Sunny, Overcast, Rain}, Humidity = {High, Normal}, Wind = {Strong, Weak} and Play = {Yes, No}; Since the datasets in a data table are not necessarily unique, we allow for multiple instances of an element existing in the same set (known as a multiset, or a bag[21]). If D T is a subset of T and q is a positive integer, then we define:

A q-multiple-of D T , denoted as q D T , is a set of datasets containing q instances of each dataset in D T .Therefore, 2 D T = { Sunny, High, Strong, Yes} , {Sunny, High, Strong, No }, {Sunny, High, Weak, Yes }, {Sunny, High, Weak, No }, {Sunny, Normal, Strong, Yes }, {Sunny, Normal, Strong, No }, {Sunny, Normal, Weak, Yes }, {, Normal, Weak, No }, {Overcast, High, Strong, Yes }, {Overcast, High, Strong, No }, {Overcast, High, Weak, Yes }, {Overcast, High, Weak, No }, {Overcast, Normal, Strong, Yes }, {Overcast, Normal, Strong, No }, {Overcast, Normal, Weak, Yes }, {Overcast, Normal, Weak, No }, {Rain, High, Strong, Yes }, {Rain, High, Strong, No }, {Rain, High, Weak, Yes }, {Rain, High, Weak, No }, {Rain, Normal, Strong, Yes }, {Rain, Normal, Strong, No }, {Rain, Normal, Weak, Yes }>, {Rain, High, Weak, No }, {Rain, Normal, Strong, Yes }, {Rain, Normal, Strong, No }, {Rain, Normal, Weak, Yes }, {□Rain, Normal, Weak, No }, {Sunny, High, Strong, Yes }, {Sunny, High, Strong, No }, {Sunny, High, Weak, Yes }, {Sunny, High, Weak, No }, {Sunny, Normal, Strong, Yes }, {Sunny, Normal, Strong, No }, {Sunny, Normal, Weak, Yes }, {Sunny, Normal, Weak, No }, {Overcast, High, Strong, Yes }, {Overcast, High, Strong, No }, {Overcast, High, Weak, Yes }, {Overcast, High, Weak, No }, {Overcast, Normal, Strong, Yes }, {Overcast, Normal, Strong, No }, {Overcast, Normal, Weak, Yes },{Overcast, Normal, Weak, No }, {Rain, High, Strong, Yes }, {Rain, High, Strong, No }, {Rain, High, Weak, Yes }, {Rain, High, Weak, No }, {Rain, Normal, Strong, Yes }, {Rain, Normal, Strong, No }, {Rain, Normal, Weak, Yes }, {Rain, Normal, Weak, No }}

We introduce, with examples, the foundations of dataset complementation and its application in decision-tree learning. The data tables in these examples have an attribute “Sample #”, which is used as a primary key reference but not as an option of a decision or test attributes.

We introduce, with examples, the foundations of dataset complementation and its application in decision-tree learning. The data tables in these examples have an attribute “Sample #”, which is used as a primary key reference but not as an option of a decision or test attributes.

B. Data Set Complement:

In this segment work is done with the sets that can contain multiple instances of the same element. The segment begins by defining fundamental concepts and then data unrealization algorithm.

a) T - Data Table

b) TS – Training Set, is constructed by inserting sample data sets into a data table.

c) TU - Universal set of data table T is a set containing a single instance of all possible data sets in data table T.

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

d) TP – Perturbed Data Set.
e) T‘ – Unrealized Training Set

C. Algorithm for Data Unrealization:

Dataset Complementation approach was designed for discrete value classification so continuous values are replaced with ranged values. The entire original dataset is replaced by unreal dataset for preserving the privacy via dataset complementation. This approach can be applied at any time during the data collection process so that privacy protection can be in effect even while samples are still being collected. The original accuracy of training dataset is preserved without linking the perturbed dataset to the information provider i.e. accurate data mining results are yields while preserving privacy of individual‘s records by dataset complementation approach. A data complementation approach requires an extra table TP for converting sample dataset Ts into an unrealized training set T‘. TP is perturbing set that generates unreal dataset.

Initially T‘ and TP are an empty set. When we get an Ts the TP is constructed with universal set TU by adding TU into TP . Whenever we get sample data item t in TS we remove it from TP and transfer one data item ti1 into T1 . Ti1 is the latest available frequent data item in TP . When traversing TP is finished and if sample data item t1 is not available in TP then To unrealized the samples TS, initialize both T‘ and TP as an empty sets, i.e. invoke the above algorithm with Unrealized_Training_set(Ts, TP, { }, { }) . The elements in the resulting data sets are unreal individually, but meaningful when they are used together to calculate the information required by a modified C4.5 algorithm [13]. 4) C4.5 Algorithm for Decision Tree Generation based on Information Entropy and Information Gain The algorithm C4.5 selects a test attribute (with the smallest entropy) according to the information content of the training set Ts . The information entropy and information gain functions are given as below. Information entropy is a term that was introduced by Claude Shannon‘s information theory in 1948. In information theory, information content is measured in bits. Entropy measures the minimum number of bits necessary to communicate information. It can also be used to measure the uncertainty associated with a random variable. If a random variable X has possible outcomes ki with probabilities ( ) i P k while i is an integer and 1 V i V n, then the information content I in bits can be expressed.

3.2 Horizontal partitioning data:

When data is horizontally partitioned, parties collect data for different entities, but have data for all of the attributes. We now need to figure out how the RDTs can be constructed and how classification is performed. Since all the parties share the schema, a straight-forward solution is for all parties to independently create a few random trees. Together these will form the ensemble of random trees. However, each party can only independently create the structure of the tree. All parties must co-operatively and securely compute the parameters (i.e., values of each leaf node), over the global data set. Unlike the basic RDT approach, there is no need to keep the class distribution at each non-leaf node—this information is only required at the leaf nodes. Now, there are two possibilities:

1) The structure of the tree is known to each participant.

2) The structure of the tree is unknown to each participant.

3.3 Vertical partitioning data:

With vertically partitioned data, all parties collect data for the same set of entities. However, each party collects data for a different set of attributes. Now the parties cannot independently create even the structure of a random tree, unless they share the attribute information among each other. Thus, there are two possibilities:

1) All parties share basic attribute information (i.e., metadata). Now they can independently create random trees (at least the structure).
2) There is no sharing of information. Now, the parties need to collaborate to create the random trees. These trees could themselves exist in a distributed form. Unlike the horizontal partitioning case, the structure of the tree does reveal potentially sensitive information, since the parties do not know what are the attributes owned by the other parties. Therefore, we directly address the case of fully distributed trees.

3.4 System Architecture:

While the use of RDTs may seem counterintuitive, there are many benefits in terms of performance and accuracy that are gained by using this method versus traditional algorithms. Fan et al. find that for classification, use of a random model can match, in terms of solutions, other inductive learning models in finding an optimal hypothesis. At the same time, RDT outperforms other models in terms of computational speed, due to the inherent properties of random partitioning used in tree construction. The RDTs algorithm builds multiple (or m) iso-depth RDTs. One important aspect of RDTs is that the structure of a random tree is constructed completely independent of the training data. The RDT algorithm can be broken into two stages, training and classification. The training phase consists of building the trees (Build Tree Structure) and populating the nodes with training instance data (Update Statistics). It is assumed that the number of attributes is known based on the training data set. The depth of each tree is decided based on a heuristic—Fan et al. show that when the depth of the tree is equal to half of the total number of features present in the data, the most diversity is achieved, preserving the advantage of random modeling. The process for generating a tree is as follows. First, start with a list of features (attributes) from the data set. Generate a tree by randomly choosing one of the features without using any training data. The tree stops growing once the height limit is reached. Then, use the training data to update the statistics of each node. Note that only the leaf nodes need to record the number of examples of different classes that are classified through the nodes in the tree. The training data is scanned exactly once to update the statistics in multiple random trees. When classifying a new instance x, the probability outputs (or regression/ranking values for regression, ranking and multi-label classification problems) from multiple trees are averaged to estimate the a posteriori probability.

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 2, Issue 08, August 2015
Available at http://internationaljournalofresearch.org

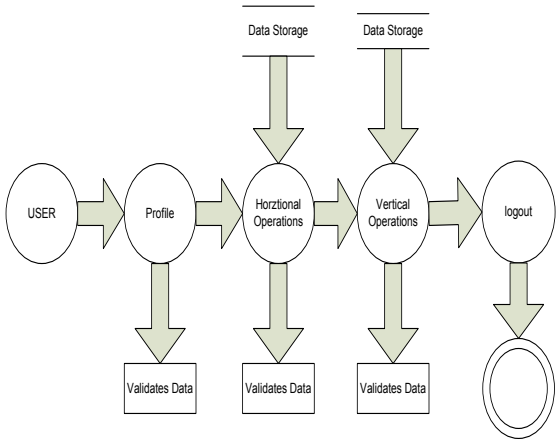

Fig 2: System Architecture Diagram:

Figure 2 shows the Architecture design of generating Random Decision trees. Data Owners stores the encrypted data in Cloud servers. Homomorphic Encryption is applied to nursery data. Only authorized users can access the data from Cloud servers. Hence the privacy of the data can be preserved.

Given a knowledge set D ≡ (T, R) distributed among k parties P1; . . . ; Pk Securely build a random decision tree classifier RDT, and provide a privacy- protective distributed classification mechanism to classify a replacement instance. There are two dependent steps to partition data: Horizontal and Vertical partitions. In this, once knowledge is horizontally divided between Pk parties, every party holds different instances, however collects a similar section of information. All parties share the schema, although the particular transactions in their native databases are distinctive. Clearly, since the schema is shared by all parties, the category attribute C is also renowned to any or all parties.

4. Experimental Work

The distributed RDT algorithms and implementation presented in this paper are a significant step forward in creating usable, distributed, privacy-preserving, data mining algorithms. The running time of the algorithms, is comparatively much faster than the existing implementations, and is usable on everyday computing hardware. As compared to the standard, non privacy-preserving version, the accuracy of the privacy-preserving solution is exactly the same, though the computational overhead is significant. However, privacy is not free. In general, privacy-preserving protocols are more expensive than non-privacy-preserving protocols for the same problem. For example, [7] shows that the privacy-preserving ID3 requires two orders of magnitude larger computation time than the non privacy preserving version. Indeed, this motivated us to build the more efficient solutions proposed in this paper, so that use of PPDM solutions can become a reality.

5. Conclusion

In this paper, we studied the technical feasibility of realizing privacy-preserving data mining. RDTs can be used to generate equivalent, accurate and sometimes better models with much smaller cost; we are using distributed privacy-preserving RDTs. Our approach leverages the fact that randomness in structure can provide strong privacy with less computation. In the future, we plan to develop general solutions that can work for arbitrarily partitioned data and overlapping transaction.

6. References

[1] S. Ajmani, R. Morris, and B. Liskov, “A Trusted Third-Party Computation Service,” Technical Report MIT-LCS-TR-847, MIT, 2001.

[2] S.L. Wang and A. Jafari, “Hiding Sensitive Predictive Association Rules,”

Proc. IEEE Int'l Conf. Systems, Man and Cybernetics, pp. 164-169, 2005.

[3] R. Agrawal and R. Srikant, "Privacy Preserving Data Mining," Proc. ACM SIGMOD Conf. Management of Data (SIGMOD '00), pp. 439-450, May 2000.

[4] Q. Ma and P. Deng, "Secure Multi-Party Protocols for Privacy Preserving Data Mining," Proc. Third Int'l Conf. Wireless Algorithms, Systems, and Applications (WASA '08), pp. 526-537, 2008.

[5] J. Gitanjali, J. Indumathi, N.C. Iyengar, and N. Sriman, "A Pristine Clean Cabalistic Foruity Strategize Based Approach for Incremental Data Stream Privacy Preserving Data Mining," Proc. IEEE Second Int'l Advance Computing Conf. (IACC), pp. 410-415, 2010.

[6] N. Lomas, "Data on 84,000 United Kingdom Prisoners is Lost," Retrieved Sept. 12, 2008, http://news.cnet.com/8301-1009_3-10024550-83.html, Aug. 2008.

[7] BBC News Brown Apologises for Records Loss. Retrieved Sept. 12, 2008, http://news.bbc.co.uk/2/hi/uk_news/politics/7104945.stm, Nov. 2007.

[8] D. Kaplan, Hackers Steal 22,000 Social Security Numbers from Univ. of Missouri Database, Retrieved Sept. 2008, http://www.scmagazineus.com/Hackers-steal-22000-Social-Security-numbers-from-Univ.-of-Missouri-database/article/34964/, May 2007.

[9] D. Goodin, "Hackers Infiltrate TD Ameritrade client Database," Retrieved Sept. 2008, http://www.channelregiste

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