PRIVACY PRESERVING DECISION TREE LEARNING USING UNREALIZED DATA SETS

R. Ch. VARMA[1] PETER PRAVEEN J[2]

[1] Asst. Professor in Visakha Institute Of Engineering And Technology,A.P,INDIA

[2] pursing M.Tech(cse) from Visakha Institute of Engineering And Technology,A.P,INDIA

ABSTRACT

In recent years, advances in hardware technology have led to an increase in the capability to store and record personal data about consumers and individuals. This has led to concerns that the personal data may be misused for a variety of purposes. In order to alleviate these concerns, a number of techniques have recently been proposed in order to perform the data mining tasks in a privacy-preserving way. These techniques for performing privacy-preserving data mining are drawn from a wide array of related topics such as data mining, cryptography and information hiding.

This approach converts the original sample data sets into a group of unreal data sets, from which the original samples cannot be reconstructed without the entire group of unreal data sets. Meanwhile, an accurate decision tree can be built directly from those unreal data sets. This novel approach can be applied directly to the data storage as soon as the first sample is collected. The approach is compatible with other privacy preserving approaches, such as cryptography, for extra protection.

Keywords: *cryptography, data mining, decision tree, security and privacy protection.*

I. INTRODUCTION

We introduce a new perturbation and randomization based approach that protects centralized sample data sets utilized for decision tree [5] data mining. Privacy preservation via dataset complementation is a data perturbed approach that substitutes each original dataset with an entire unreal dataset. Privacy preservation is applied to sanitize the samples prior to their release to third parties in order to mitigate the threat of their inadvertent disclosure or theft. In contrast to other sanitization methods, our approach does not affect the accuracy of data mining results. The decision tree can be built directly from the sanitized data sets, such that the originals do not need to be reconstructed. Moreover, 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 following assumptions are made for the scope of this technique:

First, a sufficiently large number of sample data sets have been collected to achieve significant data mining results covering the whole research target. Second, the number of data sets leaked to potential attackers constitutes a small portion of the entire sample database. Third, Identity attributes (e.g., social insurance number) are not considered for the data mining process because such attributes are not meaningful for decision making. Fourth, all data collected are discretized; continuous values can be represented via ranged value attributes for decision tree data mining

II. CONFIGURATION AND ANALYSIS

In Privacy Preserving Data Mining Models and Algorithms [6], we can classify privacy preserving data mining techniques, including perturbation-based strategies cryptographic, statistical, query auditing and data modification. Statistical, cryptographic techniques and query auditing most are subjects beyond the focus of this paper

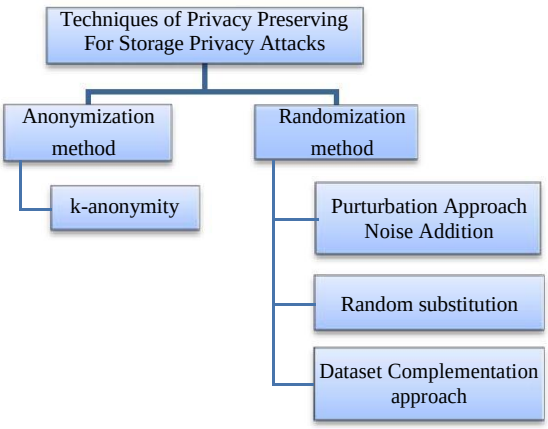


Figure 1: Techniques of Privacy Preserving for storage privacy attacks

The different privacy preserving technique for storage privacy attacks are shown in fig.1.These techniques mainly divided into two types' first Anonymization method and second Randomization method.The Anonymization operation design to prevent identification, hides some details in quasi identifier attribute and for categorical data, a specific value can be replaced with general value. The Randomization method is privacy preserving technique in which attribute values of record are masked by adding noise or by substituting random value or by using the data complementation approach. Thesetechniques are developed to derive aggregate distribution from perturbed record because noise added or substituted is sufficiently large sothat individual record values cannot be recovered.The sample dataset shown in Table 1 is used throughout the report as an example.

Dataset Complementation approach

Dataset Complementation approach [10] 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.

Table 1: Sample Dataset Ts

Outlook	Humidity	Wind	Play
Sunny	High	Weak	No
Sunny	High	Strong	No
Overcast	High	Weak	Yes
Rainy	High	Weak	Yes
Rainy	Normal	Weak	Yes
Rainy	Normal	Strong	No
Overcast	Normal	Strong	Yes
Sunny	High	Weak	No
Sunny	Normal	Weak	Yes
Rainy	Normal	Weak	Yes
Sunny	Normal	Strong	Yes
Overcast	High	Strong	Yes
Overcast	Normal	Weak	Yes
Rainy	High	Strong	No

III. DESIGN AND SPECIFICATIONS

Dataset Complementation Approach

This set work with multiple instances of the same element i.e with multisets(bags) rather than with sets as defined in the classical set theory

Universal set and Data set Complement

Definition- If data table T associates with a tuple of attributes {Wind, Play} where Wind= {Strong, Weak} and Play= {Yes, No} then T^U= {(Strong, Yes),(Strong, No),(Weak, Yes),(Weak, No)}

Remark- If data table T associates with tuple of m attributes{a_1,a_2,….a_n} where a_i has n_i possible values and $1 \leq i \leq m$ then $|T^U| = n_1 * n_2 * \ldots n_m$

Definition- If T_D is a subset of T, then the absolutecomplement of T_D, denoted as $T^C{}_D$, is equal to $T^U - T_D$, anda q-absolute-complement of T_D, denoted as $qT^C{}_D$, is equal to $qT^U - T_D$.

Example-With the conditions of Example 1, if T_D={(Strong; Yes),(Weak; Yes),(Weak,No)} then $T^C{}_D${(Strong, No)} and $2T^C{}_D$={(Strong, Yes), (Strong, No), (Weak; Yes),(Weak, No),(Strong, No)}

Unrealized Training Set

Traditionally a training set T_S is constructed by inserting a sample data set into table. However a data set complementation approach requires an extra table T^P. It is called as perturbing set that generates unreal data set which is used for converting the sample data into unrealized training set T'. The algorithm for unrealized training set is as below.

Algorithm: Unrealize- Training Set (T_S, T^U, T', T^P) **Input**:-T_S- set of input sample data set

T^U- Universal Set

T' - a set of output training data set

T^P - a perturbing set

Output- (T' , T^P)

1. It T_S is empty then return (T' , T^P)
2. t ← a data set in T_S
3. If 't' is not an element of T^P or T^P={t} then
4. T^P ← $T^P + T^U$
5. T^P ← T^P -{t}
6. t' ← the most frequent database in T^P
7. Return Unrealize- Training Set

(T_S- {t}, T^U , T' + { t'}, T^P - {t'})

Outlook	Wind	Play
Sunny	Weak	No
Sunny	Strong	No
Overcast	Weak	Yes
Rainy	Weak	Yes
Rainy	Weak	Yes
Rainy	Strong	No
Overcast	Strong	Yes

Outlook	Wind	Play
Sunny	Strong	Yes
Sunny	Strong	No
Sunny	Weak	Yes
Sunny	Weak	No
Overcast	Strong	No
Overcast	Weak	Yes
Overcast	Weak	No
Rainy	Strong	Yes
Rainy	Strong	No
Rainy	Weak	No

Outlook	Wind	Play
Sunny	Strong	Yes
Sunny	Weak	Yes
Overcast	Strong	Yes
Overcast	Strong	No
Overcast	Weak	No
Rainy	Strong	Yes
Rainy	Weak	No

(a)Ts **(b) T^P** **(c) T'**

Figure 2: Unrealizing training samples in (a) by calling Unrealize-Training-Set (TS, TU, { }, { })The resulting tables T^PandT'are given in (b) &(c).

To unrealize samples Ts, we initialize both T' & T^P as empty sets. We invoke the above algorithm with Unrealize- Training set (Ts, T^U,{},{}). The resulting unrealized training set contains some dummy data set excepting the ones in Ts. The elements in the data sets are unreal individually, but meaningful when they are together to calculate the information required by a modified ID3 algorithm.

Modified Decision Tree Generation Algorithm

As entropies of the originaldata sets, Ts, can be determined by the retrievable information i.e. the contents of unrealized training set, T', and perturbing set, T^P- the decision tree of Ts can be generated by the following algorithm.

Algorithm- Generate-Tree' (size, T', T^P ,attribs, default) **Input**: size, size of qT^U

T': the set of unreal training data sets

T^P: the set of perturbing data sets

attribs: set of attributes

default: default value for the goal

predicate **Output**: tree, a decision tree

1. if (T', T^P) is empty then return default
2. default Minority _ Value(T'+ T^P)
3. if$H_{ai}(q[T'+ T^P]^c)$= 0 then return default
4. else if attribs is empty then return default
5. else
6. best Choose-Attribute' (attribs, size,(T',T^P))

7. tree a new decision tree with root attribute best

8. size size=number of possible values k_i in best

9. for each value v_i of best do

10. T_i' ←{data sets in T0 as best = k_i}

11. T^P_i←{data sets in TP as best = k_i}

12. *subtree*←Generate-Tree(size,T_i',T_i^P,attribs-best, default)

13. connect tree and *subtree* with a branch labeled k_i

14. return tree

Fig. 3 shows the resulting decision tree of our new ID3 algorithm with unrealized sample inputs shown in Figs. 3b and 3c. This decision tree is the

Decision tree Generation - Building a Decision Tree

- First test all attributes and select the one that would function as the best root;
- Break-up the training set into subsets based on the branches of the root node;
- Test the remaining attributes to see which ones fit best underneath the branches of the root node;
- Continue this process for all other branches until
  - all examples of a subset are of one type
  - there are no examples left (return majority classification of a parent)
  - there are no more attributes left (default value should be majorityclassification)

Information Entropy and Gain

Determining which attribute is best (Entropy & Gain)

Entropy (E) is the minimum number of bits needed in order to classify an arbitrary example as yes or no

$E(S) = \Sigma^c_{i=1} -p_i \log_2 p_i$,

Where S is a set of training examples,

c is the number of classes, and

p_i is the proportion of the training set that is of class

i For our entropy equation 0 $\log_2$ 0 = 0

The information gain G(S,A) where A is an attribute G(S,A) ≡ E(S) - Σ v in values(A)(|S_v| / |S|) * E(S_v) Let's Try an Example!

Let E([X+,Y-]) represent that there are X positive training elements and Y negative elements. Therefore the Entropy for the training data, E(S), can be represented as E([9+,5-]) because of

Fig.3. Decision Tree

the 14 training examples 9 of them are yes and 5 of them are no.

Let's start off by calculating the Entropy of the Training Set.

$E(S) = E([9+,5-]) = (-9/14 \log_2 9/14) + (-5/14 \log_2 5/14)$

$= 0.94$

Next we will need to calculate the information gain G(S,A) for each attribute A where A is taken from the set {Outlook, Temperature, Humidity, Wind}.

The information gain for Outlook is:

G(S,Outlook) = E(S) – [5/14 * E(Outlook=sunny) + 4/14 * E(Outlook = overcast) + 5/14 * E(Outlook=Rainy)]

G(S,Outlook) = E([9+,5-]) – [5/14*E(2+,3-) + 4/14*E([4+,0-]) + 5/14*E([3+,2-])] G(S,Outlook) = 0.94 – [5/14*0.971 + 4/14*0.0 + 5/14*0.971]

G(S,Outlook) = 0.246

G(S,Temperature) = 0.94 – [4/14*E(Temperature=hot) +6/14*E(Temperature=mild) + 4/14*E(Temperature=cool)]

G(S,Temperature) = 0.94 – [4/14*E([2+,2-]) + 6/14*E([4+,2-]) + 4/14*E([3+,1-])] G(S,Temperature) = 0.94 – [4/14 + 6/14*0.918 + 4/14*0.811]

G(S,Temperature) = 0.029

G(S,Humidity) = 0.94 – [7/14*E(Humidity=high) + 7/14*E(Humidity=normal)]

G(S,Humidity = 0.94 – [7/14*E([3+,4-]) + 7/14*E([6+,1-])]

G(S,Humidity = 0.94 – [7/14*0.985 +

7/14*0.592] G(S,Humidity) = 0.1515

G(S,Wind) = 0.94 – [8/14*0.811 +

6/14*1.00] G(S,Wind) = 0.048

Outlook is our winner!

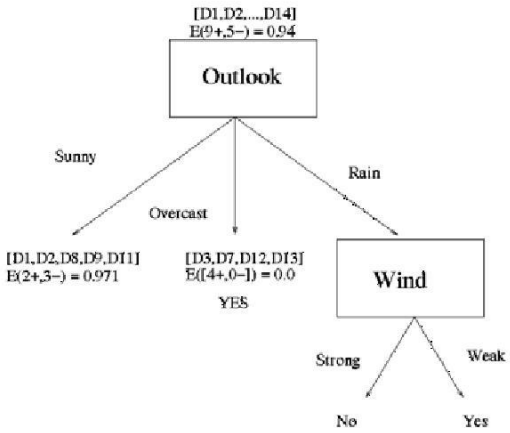
Now that we have discovered the root of our decision tree we must now recursively find the nodes that should go below Sunny, Overcast, and Rainy.

G(Outlook=Rainy,Humidity)=0.971–[2/5*E(Outlook=Rainy^Humidity = high)+ 3/5*E(Outlook=Rainy ^Humidity=normal]

G(Outlook=Rainy, Humidity) = 0.02

G(Outlook=Rainy,Wind) = 0.971- [3/5*0 +

2/5*0] G(Outlook=Rainy,Wind) = 0.971



Enhanced Protection with Dummy Values

Dummy values can be added for any attribute such that the domain of the perturbed sample data sets will be expanded while the addition of dummy values will have no impact on Ts. For example, we can expand the possible values of attribute Wind from {Strong, Weak} to {Dummy, Strong, Weak} where Dummy represents a dummy attribute value that plays no role in the data collection process shown in Fig. 3. In this way we can keep the same resulting decision tree (because the entropy of TS does not change) while arbitrarily expanding the size of T^U. Meanwhile, all data sets in T' and T^P, including the ones with a dummy attribute value, are needed for determining the entropies of $q[T'+T^P]^C$ during the decision tree generation process.

Outlook	Wind	Play
Sunny	Dummy	Yes
Sunny	Dummy	No
Sunny	Strong	Yes
Sunny	Weak	Yes
Overcast	Dummy	Yes
Overcast	Dummy	No
Overcast	Strong	No

⇩

Outlook	Wind	Play
Sunny	Weak	No
Sunny	Strong	No
Overcast	Weak	Yes
Rainy	Weak	Yes
Rainy	Weak	Yes
Rainy	Strong	No
Overcast	Strong	Yes

+

Outlook	Wind	Play
Overcast	Weak	No
Rainy	Dummy	Yes
Rainy	Dummy	No
Rainy	Strong	Yes
Rainy	Weak	No
Sunny	Dummy	Yes
Sunny	Dummy	No
Sunny	Strong	Yes
Sunny	Strong	No
Sunny	Weak	Yes
Sunny	Weak	No
Overcast	Dummy	Yes
Overcast	Dummy	No
Overcast	Strong	Yes
Overcast	Strong	No
Overcast	Weak	Yes
Overcast	Weak	No
Rainy	Dummy	Yes
Rainy	Dummy	No
Rainy	Strong	Yes
Rainy	Strong	No
Rainy	Weak	No

Fig.4. Enhanced Protection with Dummy Values

Data Set Reconstruction

We have introduced a modified decision tree learning algorithm by using the unrealized training set, T', and the perturbing set, T^P. Alternatively, we can reconstruct the original sample data sets, Ts, from T' and T^P followed by an application of the conventional ID3 algorithm for generating the decision tree from Ts. The reconstruction process is dependent upon the full information of T' and T^P reconstruction of parts of Ts based on parts T' and T^P is not possible.

IV. RESULTS AND DISCUSSION

Privacy in Data Set Complementation

For dataset complementation, the unrealized training set *T*'and perturbing set $T^P$is introduced. A dataset in (T'+T^P) does not have any direct relationship withany dataset in the sample datasets in *S T* . Datasets in *S T* should be recovered as a wholeby using all of the datasets in *T"* and T^P.

Privacy Loss on Low Variance Cases

we found that privacy preservation of the sample datasets relieson |Ts| , $|T^U|$, $|T^P|$ and $Var(T^S)$. While |Tu| is a constant because it is based on thedomain of a sample dataset, |Ts|is dependent upon the statistical purposes and $|T^P|$ isproven as controllable, $Var(T^S)$is an uncontrollable variable that drives the privacypreserving result, whereas a higher variance preserves more privacy.

Complexity

For the function Unrealized Training-Set the timecomplexity is O(Ts). However, we need to prepare a universal set as an input of thisfunction.thisimproved approach results in a matching rate that is always less than one-third of the bestcase of the unprotected samples. In all cases, the complexity of the sanitization process isO(Ts). However, the worst case requires (2*|T^U|-1) times the amount of storageneeded for unprotected samples.

Output Accuracy

In all cases, the decision tree(s) generated from theunrealized samples (by algorithm Generate-Tree' is describedabove) is the same as the decision tree(s), TreeTs,generated from the original sample by the regular method.This result agrees

Limitations

Privacy preservation via data set complementation fails if all training data sets are leaked because the data set reconstruction algorithm is generic. Therefore, further search is required to overcome this limitation.

The worst case happens when the samples are distributed extremely unevenly. Based on the randomly picked tests, the storage requirement for our approach is less than five times (without dummy values) and eight times (with dummy values, doubling the sample domain) that of the original samples.

VI. CONCLUSION

This thesis presents a new privacy preserving approach via datasetcomplementation, which removes each sample from a set of perturbing datasets. During the privacy preserving process, this set of perturbed datasets is dynamicallymodified. As the sanitized version of the original samples, these perturbed datasets arestored to enable a modified decision tree datamining method. This method guarantees to

provide the same datamining outcomes as the originals, which is proved mathematicallyand by a test using one set of sample datasets in this thesis.

This improved approach results in a matching rate that is always less than one-third of the best case of the unprotected samples. In all cases, the complexity of the sanitization process is O(|Ts|) However, the worst case requires (2*|T^U| -1) times the amount of storage needed for unprotected samples Future research should also explore means to reduce the storage requirement associated with the derived dataset complementation approach. This technique relies on theoretical proofs with limited practical tests, so testing with real samples should be the next step to gain solid ground on real-life application.

As it is very straightforward to apply a cryptographic privacy preserving approach, such as the (anti)monotone framework, along with data set complementation, this direction for future research could correct the above limitations.

Future research should develop the application scope for other algorithms, such as C4.5 and C5.0, and data mining methods with mixed discretely-and continuously valued attributes. Furthermore, the data set complementation approach expands the sample storage size (in the worst case, the storage size equals (2|T^U-1|*|Ts|) ; therefore, further studies are needed into optimizing 1) the storage size of the unrealized samples, and 2) the processing time when generating a decision tree from those samples.

REFERENCES

[1] R. Agrawal and R. Srikant, “Privacy Preserving Data Mining,” Proc. ACM SIGMOD Conf. Management of Data (SIGMOD ’00), pp. 439-450, May 2000.

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

[3] L. Liu, M. Kantarcioglu, and B. Thuraisingham, "Privacy Preserving Decision Tree Mining from Perturbed Data," Proc. 42nd Hawaii Int'l Conf. System Sciences (HICSS '09), 2009.

[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] LiorRokach and OdedMaimon "Top-Down Induction of Decision Trees Classifiers – A Survey" IEEE Transactions On Systems, Man And Cybernetics: Part C, Vol. 1, No. 11, November 2002

[6] C. Aggarwal and P. Yu, Privacy-Preserving Data Mining:, Models and Algorithms. Springer, 2008.

[7] L. Sweeney, "k-Anonymity: A Model for Protecting Privacy," Int'l J. Uncertainty, Fuzziness and Knowledge-based Systems, vol. 10, pp. 557-570, May 2002.

[8] J. Dowd, S. Xu, and W. Zhang, "Privacy-Preserving Decision Tree Mining Based on Random Substitions," Proc. Int'l Conf. Emerging Trends in Information and Comm. Security (ETRICS '06), pp. 145-159, 2006.

[9] Mohammad Ali Kadampur, Somayajulu D.V.L.N. "A Noise Addition Scheme in Decision Tree for Privacy Preserving Data Mining" Journal of Computing, Vol. 2, Issue 1, Jan. 2010, ISSN 2151-9617

[10] Pui K. Fong and Jens H. Weber-Jahnke, "Privacy Preserving Decision Tree Learning Using Unrealized Data Sets" ." IEEE Transl. on knowledge and data engineering, vol. 24, no. 2, February2012.

[11] Wei Peng, Juhua Chen and Haiping Zhou "An Implementation of ID3 - Decision Tree Learning Algorithm" Project of Comp 9417: Machine Learning University of New South Wales, School of Computer Science & Engineering, Sydney, NSW 2032, Australia

PETER PRAVEEN J Working as Asst. Professor in Visakha Institute Of Engineering And Technology, Narava. And I Worked as a Asst.Prof in Thandra Paparaya Institute of science &Technology from 01-07-2005 to 31-10-2007. I obtained M.Tech (CSE) from Gayatri Vidhya Parishad College of engineering. M.Sc (Applied Mathematics) from Pydah College.

R. Ch. VARMA pursing M.Tech(cse) from Visakha Institute of Engineering And Technology, Narava,Visakhapatnam . I obtained **MCA** From Pyadah **P.G College.**

