ISSN:0975-9646

Preservation of the Sample Data with Help of Unrealized Training Datasets and later classifying it Using Modified C4.5 Algorithm

Kanani Arjun

Information Technology, Parul Institute of Engineering and Technology
Vadodara, India

***Abstract* - In order to protect the data centrally when they are being transferred from one party to another party so, that it cannot be used for secondary purposes unrealized training dataset is an important technique used to prevent data. With help of Unrealized training dataset algorithm it divides the sample data in two forms i.e. Tp a set of perturbing datasets and T' a set of output training datasets. The classification method used over here is C4.5 and C4.5 is the extension version of ID3 algorithm. C4.5 is the classification decision tree algorithm which uses features like handling both continuous and discrete attributes, handling missing values, purning techniques. This paper produces a modified C4.5 algorithm which uses datasets generated by unrealized training dataset for classification. As the memory consumption and time consumption rate of C4.5 is better compared to ID3 which is useful during large dataset entries to securely transfer and regenerate original data from modified C4.5 classification method.**

***Keywords* – Decision tree algorithm, ID3 and C4.5 classifier, unrealized training datasets.**

I. INTRODUCTION

Data mining is the way through which we can extract hidden information from large amount of datasets. Data collected from one party and delivered to another party is important to protect centrally as it can be used for decision making or pattern recognition. The privacy preserving technique used over here is data modification and perturbation based approaches and then classifying it using decision tree algorithm. Classification technique used over here is very unique and efficient in data mining.

Decision tree method C4.5 is the extended version of ID3 as it can handle both continuous and discrete attributes, missing values, attributes with different costs and pruning method.

a. Problem Definition

- It introduces a privacy preserving approach which can be applied to decision tree algorithm, without related loss of accuracy.
- It describes an approach to the preservation of the privacy of collected data samples in cases where information from the sample database has been partially lost when it is being transferred from one party to another.
- It converts the original sample data sets into a group of unreal data sets, from where the original samples cannot be reconstructed without the entire group of unreal data sets.
- It uses classification method C5.0 to generate decision tree from unrealized dataset.

b. Objective

The main objective behind using C4.5 algorithm was less consumption of the memory space and time during execution as the dataset to be tested are very large and on applying modified unrealized training dataset algorithm it consumption is going to nearly double as it produces both the perturbing dataset(Tp) and unrealized dataset(T'). C4.5 will help to produce smaller decision tree comparatively and features like handling both continuous and discrete attributes, handling missing values, purning techniques are also available.

c. Scope of the Work

As the further scope of the research work implementation can be done for the new features like: Feature Selection, Reduced Error Pruning, Cross Validation, Winnowing, boosting and Model Complexity by implementing the diversities of algorithm using RGUI with weka packages, the classification accuracy can be improved. However, the performance can vary with the variation in the datasets to be used as test cases. The results needs verification on huge database and since generated perturbed dataset increases exponentially with the increased size of Universal dataset so, optimization of size of perturbed datasets is an issue to resolve which could be possible by using different cryptographic methods.

II. SURVEY RELATED TO WORK DONE

a. Data Mining

Data mining is the process of finding different patterns and knowledge from large amount of data been collected from various databases and data warehouses. Now-a-days data mining is widely used by researchers for science and business process. This data is collected from individuals that are also refereed as information providers play an important for pattern reorganization and decision making. This type of data collection process takes time and efforts hence sample datasets are sometime stored for later reuse of them when needed. However third party attacks are attempted to steal these sample datasets and private information may be leaked from these stolen datasets. Therefore in order to protect these data from them being misused privacy preserving data mining techniques are developed to convert sensitive datasets into sanitized version in which private or sensitive information is hidden from unauthorized retrievers i.e. the unauthorized access.

b. Data Preservation Technique

In order to preserve centralized data from unauthorized access the technique being used over here is the perturbation-based approaches which attempt to achieve privacy by distorting information from the original data sets. It is done in such a manner that even after data been distorted perturbed data sets still retain features of the originals so that they can be used to perform data mining operations directly or indirectly via data reconstruction. Random substitutions is a perturbation approach used over here that randomly substitutes the values of selected attributes to achieve privacy protection for those attributes, and then applies data reconstruction methods when these data sets are needed for data mining.

C. Unrealized Training Dataset Approach

In order to protect data and reconstruct original data from randomized data unrealized training dataset approach is been used. Firstly, a training set T is constructed by inserting sample datasets into a data table. Secondly, a data complementation approach requires an extra data table Tp where Tp is a perturbing set that generates unreal datasets for converting the sample datasets as an unrealized training set T‟. Then whenever we get a sample dataset t, we remove t from the perturbing table Tp if Tp contains t and Tp\ {t} is not empty; otherwise, we insert {t}c = Tu\{t} into Tp. After that, one dataset t'i is transferred from Tp to T'. For convention, the first dataset in Tp will be transferred to T' over here. If Ts = {t1, t2,..,tn} is the set of sample datasets taken in the sample collection process and ti is a dataset we get from Ts each time, then the procedure for generating the unrealized training set T' from the set of sample datasets Ts can be described as follows:

```
UNREALIZED TRAINING-SET (Ts, TU, T‟, Tp) returns <T‟, Tp>
Inputs: Ts, a set of input sample datasets
TU, a universal set
T‟, a set of output training datasets
Tp, a set of unreal datasets
if Ts is empty then
return <T', Tp >
ti ← a dataset in Ts
if ti is an element of Tp and Tp \ {ti} is not empty then
Tp ← Tp – {ti}
t‟i ← a dataset in Tp
else
Tp ← Tp + TU – {ti}
t’i ← a dataset in Tp
return UNREALIZED TRAINING-SET( Ts – {ti}, TU, T‟+{t‟i}, Tp - {t‟i})
```

1) Terminal Case: if Ts = {}, return training set T' and perturbing set Tp.

2) Recursive Case #1: if ti ∈ Tp and Tp\{ti}≠{}, then Tp = Tp – {ti} – {ti‟}, Ts = Ts – {ti}and T‟ = T‟ + {ti‟}. Build T' from Ts with Tp .

3) Recursive Case #2: if ti ∉ T' or Tp\{ti}={}, then Tp = Tp + {ti}c – {ti‟} , Ts = Ts – {ti}and T‟ = T‟ + {ti‟}. Build T' from Ts with Tp .

Pseudocode for unrealizing the training set is shown below. To unrealize the samples, we initialize both T' and Tp as empty sets, i.e. UNREALIZED TRAINING SET (Ts , TU , {}, {}) is called and then above rules are been followed. UNREALIZED TRAINING-SET (Ts, TU, T‟, Tp) returns <T‟, Tp>

d. Decision trees

A decision tree describes sequential tests and their corresponding test outcomes where test input is represented by a set of attributes with values. The outcome, which is known as the decision, represents the predicted output values of the input. The values of the inputs and outputs can be discrete or continuous. Regression learning approximates continuous-value functions and classification learning approximates discrete value functions. In this thesis, we are focusing on classification learning, while continuous values can be treated as discrete by applying value ranges instead. Decision-tree structure can be used to represent meaningful information for humans, such as instructions and manuals; therefore, it is a common class of inductive learning methods.

Decision tree structures are a most common way for classification. Classification using a decision tree is performed by moving in top to down approach i.e. from the root node until arriving at a leaf node. The research work is made up from three different classification algorithms ID3, C4.5 and C5.0 In many applications, rulesets are preferred because they are simpler and easier to understand than decision trees. Both C4.5 and C5.0 can produce classifiers expressed either as decision trees or rulesets, but C4.5's ruleset methods are both time and memory consuming.

Psuedocode of Original C4.5[1]

```
Inputs: T: the training set.
A: the list of attributes.
y: the target attribute.
default: the default value.
Tree: fully grown tree
Outputs: D: a decision tree.
Algorithm: C4.5-GenTree (T, A, y, default)
Tree d = new Tree;
if (|Ts| = 0 or |A| = 0)
{
d.type = leaf;
d.class = default;
return d;
}
default ← Majority Value (T);
if (H(y, Ts) = 0)
{
d.type = leaf;
d.class = default;
return d;
}
Attribute b ← BestsplitAttribute (A, T);
d.type = internal;
d.class = b;
for each split of b:
{
        Tree s = C4.5-GenTree$\sigma\emptyset(b)$ (Ts ),A - {b}, size/dom(b),default);
        d.AddSubtreeWithLabel (s, ∅ (b));
    }
    return d;
Prune Tree (Tree);
```

III. COMPARISON BETWEEN EXISTING SYSTEM

.C4.5 Improvements from ID3 algorithm

1. How C4.5 handles both continuous and discrete attributes - In order to handle continuous attributes, C4.5 creates a threshold and then splits the list into those whose attribute value is above the threshold and those that are less than or equal to it.

2. How C4.5 handles training data with missing attribute values - C4.5 allows attribute values to be marked as „?‟ for missing. Missing attribute values are simply not used in gain and entropy calculations.

3. Handling attributes with differing costs.

4. Using Pruning trees after creation - C4.5 goes back through the tree once it's been created and attempts to remove branches that do not help by replacing them with leaf nodes.

IV. PROPOSED SYSTEM AND IT'S COMPARISON WITH IMPROVED ALGORITHM

Proposed Algorithm[1]

Inputs parameters:

T' : the unreal training set.
Tp : the unreal perturbing set.
A: the list of attributes.
size: the size of q(U) for these unreal sets.
default: the default value.
Tree: fully grown and unpruned tree

Output parameter: D: a decision tree.

Algorithm: C4.5 MGenTree (T', Tp, A, size, default)

```
Tree d = new Tree;
if (|T‟ ∪ TP | = 0 or |A| = 0)
{
    d.type = leaf;
    d.class = default;
    return d;
}
default Minority Value (T'∪ TP); //Modified
if (H(y, q(U) - (T' ∪ TP )) = 0) //Modified
{
    d.type = leaf;
    d.class = default;
    return d;
}
Attribute b BestsplitAttributeUnreal (A, size, (T' ∪ TP )); // Modified
d.type = internal;
d.class = b;
for each split of b:
{
    Tree s = C4.5-MGenTree ($\sigma\emptyset(b)$ (T' ∪ TP ),A - {b}, size/dom(b),default); //
    Modified
    d.AddSubtreeWithLabel (s, ∅ (b));
}
return d;
Prune Tree (Tree);
```

Table 4.1 Performance comparison between training datasets of weather [1]

Performance Measure	On Real Datasets using C4.5		On Unreal Datasets using Modified C4.5	
True Positive	Class yes	Class No	Class Yes	Class No
	0.667	0.4	0.7	0.15
False positive	0.60	0.333	0.843	0.297
Precision	0.667	0.4	0.49	0.297
F – Measure	0.667	0.4	0.576	0.199

Table 4.2 Performance comparison between test datasets of weather [1]

Performance Measure	On Real Datasets using C4.5		On Unreal Datasets using Modified C4.5	
True Positive	Class yes	Class No	Class yes	Class No
	1	0.167	0.75	0.1
False positive	0.883	0	0.167	0.429
Precision	0.444	1	0.75	0.5
F – Measure	0.615	0.286	0.75	0.667

Limitations of Existing System

In the existing system memory (during ruleset generation) and time consumed by modified ID3 decision tree algorithm was more comparatively which was improved by use of modified C4.5 and these could be still improved by using C5.0. Even the accuracy rate and size of decision tree can be improved using C5.0 as it provides boosting facility.

Experimental Parameters

Speed - C5.0 is significantly faster than C4.5 (several orders of magnitude)

Memory usage - C5.0 is more memory efficient than C4.5. C5.0 commonly uses an order of magnitude less memory than C4.5 during ruleset construction.

Accuracy: The C5.0 rulesets have noticeably lower error rates on unseen cases. Sometimes the C4.5 and C5.0 rulesets have the same predictive accuracy, but the C5.0 ruleset is smaller.

Smaller decision trees - C5.0 gets similar results to C4.5 with considerably smaller decision trees.

Support for boosting - Boosting improves the trees and gives them more accuracy.

Weighting - C5.0 allows you to weight different attributes and misclassification types.

Winnowing - C5.0 automatically winnows the data to help reduce noise.

V. CONCLUSION AND FUTURE WORK

The survey done till now concludes that universal datasets can be divided into unreal datasets using unrealized training dataset method with help of sample dataset collected from universal datasets. It can be used for centrally preventing data when passed from one party to another. The data obtained can easily be re-obtained by using classification methods like Id3, C4.5 and C5.0. Out of Id3 and C4.5 classification methods C4.5 shows better results compared to ID3 and these can also be improved by using C5.0 algorithm as still the space and time consumption issues are needed to be taken care of and accuracy can also be improved. We can use winnowing and boosting methods to improve over results and can also solve the consumption issues.

REFERENCES

[1] Privacy Preserving Classification By Using Modified C4.5, Ranjan Baghel Department of Computer Science & Engineering, National Institute of Technical Teachers Training & Research, Chandigarh, India and Maitreyee Dutta Department of Computer Science & Engineering, National Institute of Technical Teachers Training & Research, Chandigarh, India, IEEE International conference on Data Mining (ICDM), August 2013.

[2] Improved Decision Tree Induction Algorithm with Feature Selection, Cross Validation, Model Complexity and Reduced Error Pruning, A. S. Galathiya, C. K. Bhensdadia, Faculty of Technology, D. D. University – Nadiad, India and A. P. Ganatra, Charotar Institute of Technology- Changa, India, International Journal of Computer Applications (0975 – 8887) Volume 46– No.23, May 2012.

[3] Privacy Preserving Decision Tree Learning Using Unrealized Data Sets, Pui K. Fong and Jens H. Weber-Jahnke, Senior Member, IEEE Computer Society IEEE Transactions on knowledge and data engineering, Vol. 24, No. 2, February 2012.

[4] Classification with an improved Decision Tree Algorithm, A. S. Galathiya, C. K. Bhensdadia, Faculty of Technology, D. D. University – Nadiad, India and A. P. Ganatra, Charotar Institute of Technology- Changa, India, International Journal of Computer Applications (0975 – 8887) Volume 46– No.23, May 2012.

[5] Are Decision Trees Always Greener on the Open (Source) Side of the Fence?, Samuel A. Moore, Daniel M. D'Addario, James Kurinskas, and Gary M. Weiss, Department of Computer and Information Science, Fordham University, Bronx, NY, USA.

[6] Efficient decision tree construction in unrealized dataset using C4.5 algorithm, A.P.Subapriya, PG Student, Department of IT, SNS College of Technology, Coimbatore, Tamil Nadu, India, M.Kalimuthu, Associate Professor, Department of IT, SNS College of Technology, Coimbatore, Tamil Nadu, India , Research Journal of Computer Systems Engineering – RJCSE, Vol 04; Special Issue; June 2013.

[7] Application of Data Mining Technique for Diagnosis of Posterior Uveal Melanoma, Darius, Arunas Institute of Biomedical Engineering, Kaunas University of Technology Student, 50–343, LT-3031 Kaunas, Lithuania e-mail: arunas.lukosevicius@ktu.lt, Alvydas, Valerijus Department of Ophthalmology, Institute for Biomedical Research Kaunas University of Medicine Eiveniu 4, LT-3007 Kaunas, Lithuania e-mail: apaun@medi.lt, INFORMATICA, 2002, Vol. 13, No. 4, 455–464 455, Institute of Mathematics and Informatics, Vilnius, August 2002.