International Journal of Computational Intelligence and Informatics, Vol. 4: No. 2, July – September 2014

Binary Decision Tree Classification based on C4.5 and KNN Algorithm for Banking Application

J Chitra Devi
Department of Computer Technology
Anna university, MIT campus
Chennai, India
chitradevi21@gmail.com

***Abstract*- In current era, database is widely used for storage purpose. History of these data which are stored in database and data warehouse has to be used optimally to analysis and predict the current trends. Mining of the data is required to perform the analysis. Data mining extracts the knowledge from the database or data warehouse. Extracted knowledge is represented in the form of various models. Among the models in data mining, Classification is the widely used model of data representation. To classify the data in the dataset, decision tree approach is introduced. There are various algorithms introduced in data mining technologies of which C4.5 identified to be a famous algorithm. C4.5 classifier uses the information gain ratio as the parameter to build the decision tree. Rules are extracted from the decision tree. Inconsistencies in the dataset are also considered before decision tree is built. KNN algorithm is used to resolve the inconsistencies due to missing data. This KNN algorithm is a clustering based approach. Size of the decision tree will depends on the attribute types at each node namely categorical or numerical. Thus split at a node will lead to numerous child nodes depending upon the type of attributes at the node to be split. This paper proposes a binary decision tree construction irrespective of attribute types. Thus the project always built the binary decision tree with only two splits at each attribute. The modified decision tree also proves the efficiency in terms of true positive rate and false positive rate compared to initial decision tree.**

Keywords- Decision Tree, Data imputation, KNN, Data Preprocessing, Knowledge Base

I. INTRODUCTION

Data mining is the wide area where the dataset under experiments are analyzed and patterns are extracted. Dataset is the collection of attributes and its value for various instances. Attributes represent the characteristics of the data and instances are the values of the data. It can be represented as data matrix of form n x m, where n represents number of instances and m represents attribute count. Each dataset has a attribute of interest that need to be identified. These attribute is termed as the class label. Data mining techniques work with dataset to receive the outcome. Outcome can be a decision tree model, rules mining, data classification etc. Various widely used techniques in data mining are Association rules mining, Classification, Clustering, Temporal mining, Spatial mining etc. Among the techniques, Classification plays a crucial role in data mining. Neural network based classification, Decision tree, Likelihood analysis, Naïve Bayes classification are few classification techniques employed. In this paper, decision tree based classification is taken into account.

The decision tree classification deals with the construction of decision tree with internal nodes and leaf node. Each internal node represents the attribute and its value leading to decision making process. Leaf node represents the class label. Each of the leaf indicates the rule in the decision tree. The decision tree is built based on few of the classification algorithms namely ID3, C4.5, CART etc. ID3 algorithm uses information gain formulae as quoted by Quinlan [1]. But this ID3 approach fails to work with numerical data in the dataset. Thus C4.5 classifier came to the picture. C4.5 classifier algorithm deals with numerical attributes as well as categorical attributes. The decision tree formed is a general tree. In C4.5, for each value of the categorical attributes of the dataset, a decisive node is created and for numeric attributes, a split location is chosen with the predefined threshold. This split point is adjusted in such a way that its attribute value and class label varies. In C4.5, for categorical attributes, the node of child node formed depends on the possible categorical attribute values whereas for numerical attributes, only binary split on the attribute at the split point is possible. The binary split based on split point value leads to two child, left and right child. Left child holds the value less than or equal to split point and right child is higher than split point value. The raw dataset might have several inconsistencies namely missing data, noisy data, outliers etc. ID3 algorithm does not deal with these inconsistencies in the dataset. A complete dataset is required in ID3 whereas C4.5 can work with the dataset with missing data values. For decision tree construction, the C4.5 classifier deletes the instances whose attribute values are missing. C4.5 classifier is capable of tackling both numeric and categorical attribute. If categorical

attribute has n different values, then the split at the attribute will leads to n distinct nodes. If n is large, the decision tree will have high complexity. Thus reduction in decision tree size is required. CART algorithm always produces the binary decision tree, but uses the permutation on categorical attribute to split the node.

II. RELATED WORK

Quinlan J. R. et al [1] resolved the C4.5 application on continuous attribute. The work proves that the decision tree could have high accuracy in data classification with inclusion of decision tree. In this paper, Continuous attribute is split based on the cut point identified. The cut point is chosen in such a way that the attribute value and its class label varies. If either attribute value or its class label identified to be the same, then the cut point is moved a step ahead. Once the condition is met, the gain ratio for the attribute is calculated at that point. Thus it always gives the binary split or ternary split at a node. Binary split is with the value less than or equal to and greater than. Ternary split is with less than, equal to and greater than of attribute value. Harvinder Chauhan et al [2] propose the implementation of decision tree algorithm C4.5 in weka software. Though this implementation, he has collected various data in a statistical manner along with the visually built decision tree. Kalpesh et al [3] compared the performance of ID3 and C4.5 classification algorithms for the student's dataset. C4.5 algorithm proves to be high efficient and accurate compared to ID3 algorithm.

III. SYSTEM DESIGN

A. Proposed System Architecture

The proposed system architecture of this project is as shown in the Fig. 1. The components of the proposed system design are composed of data mart or data warehouse, Data preprocessing, Data Classification, Knowledge base. The detailed description of the components is listed in the next section.

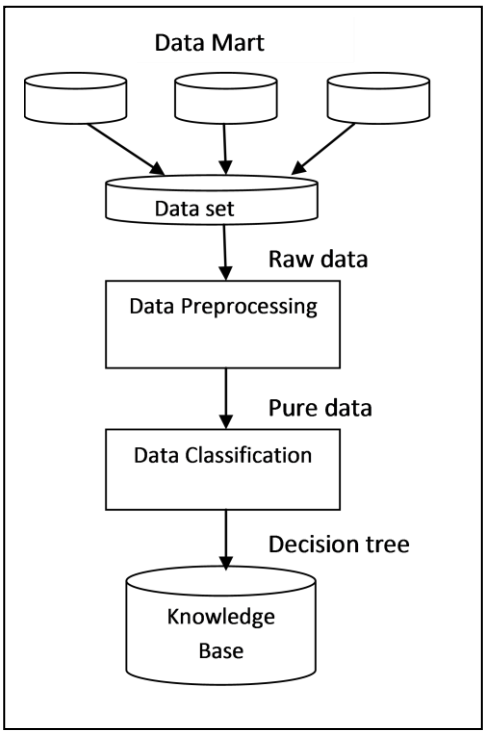

Figure 1. Proposed System Architecture

Dataset collected in the data mart is given as input to the C4.5 classifier. The C4.5 classifier constructs the decision tree based on gain ratio as given by Quinlan [1]. The built tree undergoes optimization process using Q-Learning technique. The extracted rules are stored in the knowledge base.

B. Components of the proposed Architecture

1) Data mart

Data marts are the repository of data from different source. It might be located any of the areas in the world connected via internet. The dataset are formed by concatenating the data from various source over a period of time. These dataset is given as input to the C4.5 classifier.

2) Data Preprocessing

Various Data preprocessing techniques are available to preprocess the data. Various inconsistencies in the raw dataset are missing data, noisy data, outliers etc. Missing data are imputed based on a clustering techniques

namely K-Nearest Neighbour algorithm over median computation. The distance between the dataset are measured based on Euclidean formula (1)

$$\boldsymbol{distance} = \sqrt[2]{(\boldsymbol{x1} - \boldsymbol{y1})^2 + \cdots + (\boldsymbol{xn} - \boldsymbol{yn})^2} \quad (1)$$

Where, $x_1, x_2, ..x_n$ and $y_{1,y2,} \ldots y_n$ are data points.

Another type of data preprocessing performance in the raw dataset is data transformation. The categorical attributes are identified in the dataset. The labels of the categorical attributes are enumerated starting from number 1. In the dataset, for each label, the dataset is transformed to its enumeration. Thus the resulting dataset is the pure numerical dataset.

3) Data Classification

Data Classification model constructs the decision tree based on C4.5 algorithm as defined by Quinlan [1]. The formula to calculate gain ratio are given below.

$$\boldsymbol{E(S)} = -\boldsymbol{p_i log p_i} \quad (2)$$

Where, i – 1 .. count of class labels

P_i - probability of occurrence of class label in dataset.

$$\boldsymbol{I(S,A)} = \sum_i^n \frac{|S_i|}{|S|} \boldsymbol{E(S_i)} \quad (3)$$

$$\boldsymbol{Gain(S,A)} = \boldsymbol{E(S)} - \boldsymbol{I(S,A)} \quad (4)$$

$$\boldsymbol{Split(S,A)} = -\sum_i^n \frac{|S_i|}{|S|} \mathbf{log}\left(\frac{|S_i|}{|S|}\right) \quad (5)$$

$$\boldsymbol{GainRatio(S,A)} = \frac{\boldsymbol{Gain(S,A)}}{\boldsymbol{Split(S,A)}} \quad (6)$$

4) Knowledge Base

It is used to store the rules extracted from the decision tree. Rules are of if-then format. Starting from the root node of decision tree to each leaf node, new rules are generated and added to the Knowledge base.

IV. IMPLEMENTATION DETAILS

A. Data preprocessing

The two data preprocessing steps are considered namely

1. Data imputation
2. Data transformation

The working of this data preprocessing step is shown in the flow chart as below Fig. 2.

1) Data imputation

The process of filling the missing values in the dataset is defined as data imputation. The missing values are computed based on median computation of k-nearest neighbor algorithm with Euclidean computation of distance is considered.

2) Data transformation

The process of transforming the data of one form to another is data transformation. In this paper, data is transformed from categorical attribute value to numerical attribute value. Thus the resultant dataset is of only numeric value instead of mixed dataset.

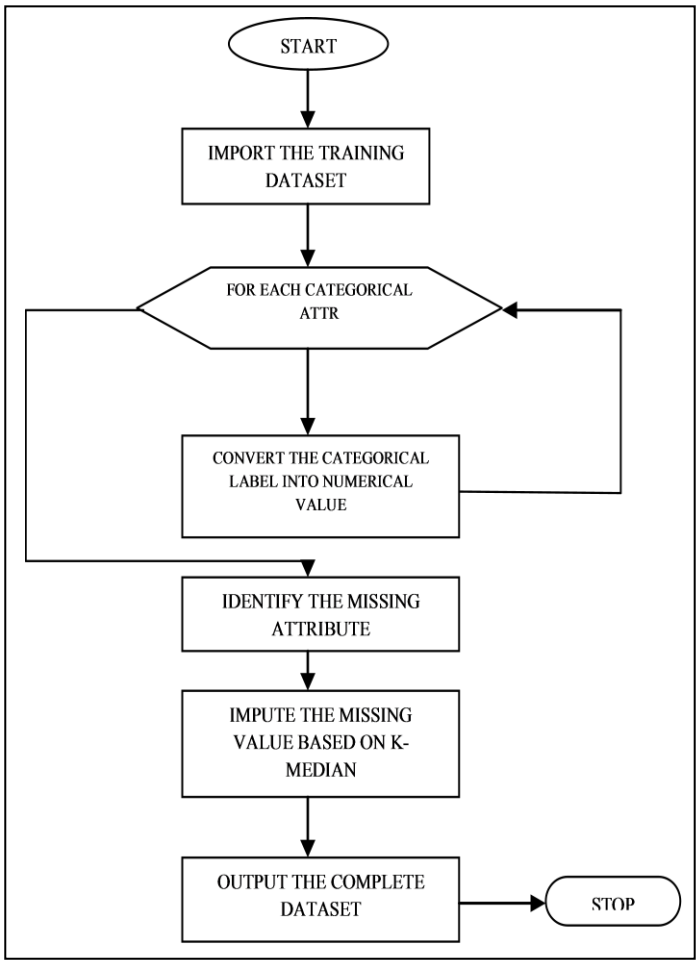

Figure 2. Flowchart of data pre-processing

B. Data classification

The algorithm for data classification is designed as below according to Quinlan's formulae [1].

ALGORITHM FOR BINARY DECISION TREE CONSTRUCTION

//Input: preprocessed dataset , //output: decision tree

1. initialize Tree = $\emptyset$
2. for each attribute a $\in$ A
2.1. identify the cut point where the attribute value and its target before and after the cut point varies.
2.2. calculate information gain ratio on cut point,

$$\mathbf{GainRatio(S,A)} = \frac{\mathbf{Gain(S,A)}}{\mathbf{Split(S,A)}}$$

3. endFor
4. choose the attribute based on maximum gain ratio
5. add the selected attribute to the node in Tree T
6. form sub dataset S
7. If target of the dataset S is same then
7.1 backtrack the decision tree and find the node whose subset S has the different class label
7.2. if no such node found then
7.2.1. return Tree
7.3. endIf
8. endIf

Figure 3. Algorithm for binary decision tree construction

V. EXPERIMENTAL RESULTS

The experiment is performed on the banking dataset. It is a dataset composed of 17 attributes and 45211 instances. The details of the attributes are listed in the Table 1.

TABLE 1. DATASET DESCRIPTION SOURCE

Attribute Name	*Attribute Type*
Age	Numeric
Job	Categorical
Marital status	Categorical
Education	Categorical
Default	Categorical
Balance	Numeric
Housing	Categorical
Loan	Categorical
Contact	Categorical
Day	Numeric
Month	Categorical
Duration	Numeric
Campaign	Numeric
Pdays	Numeric
Previous	Numeric
Poutcome	Categorical
Status	Categorical

The performance of the proposed algorithm is measured and ROC (Receiver Operating Characteristics) graph is plotted to confirm the efficiency. This ROC graph is a plot between false positive rate verses true positive rate. The coordinates (0,1) specifies that false positive rate is 0, i.e. misclassification is null and data are perfectly classified. The point (0,0) represents a classifier that predicts all cases to be negative-misclassified for all dataset. The point (1,1) indicates that all dataset are classified as positive irrespective of expected labels. Coordinate(1,0) is the incorrect classification of all cases. Thus false positive rate(FP) and true positive rate(TP) is used to show the performance of the proposed algorithm. It is shown in the Fig. 4.

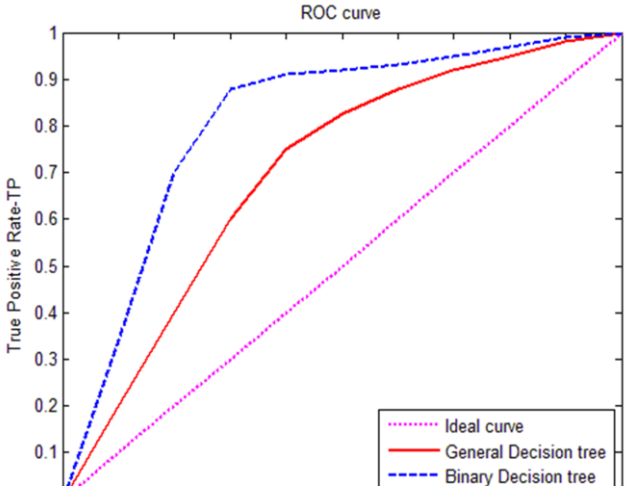

Figure 4. ROC curve

The decision built with data transformation and data imputation is shown in the Fig. 5.

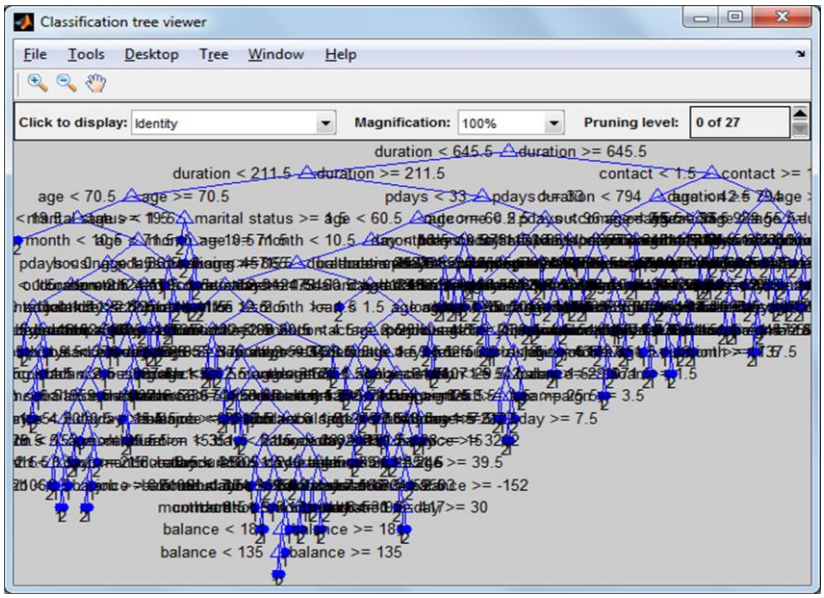

Figure 5. Decision tree with preprocessing

The part of decision tree is magnified in Fig. 6.

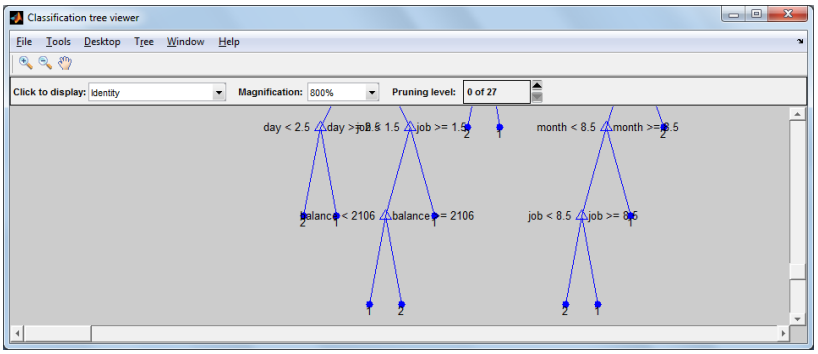

Figure 6. Magnified decision tree

The decisive rules that are stored in the knowledge base are shown in the Fig. 7.

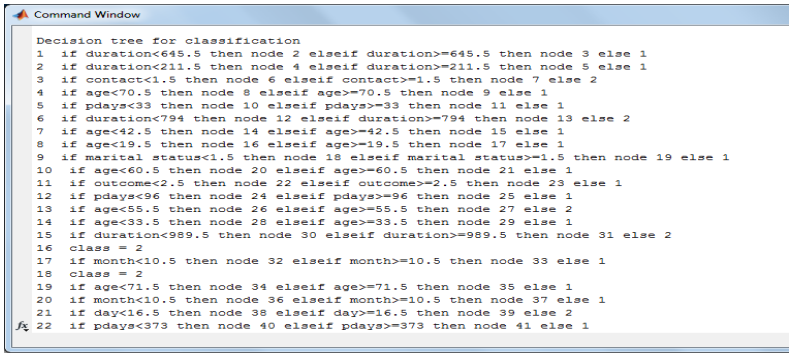

Figure 7. Decisive rules in Knowledge base

VI. CONCLUSION

The proposed algorithm is efficiently designed to build binary decision tree. ROC graph of the experiment proves the same. The test dataset are also classified properly when built tree based on pre-processed data is used for data classification. This paper concludes that the binary decision tree classification based on C4.5 classifier is far superior compare to the general decision tree built.

REFERENCES

[1] Quinlan, J. R, "Improved Use of Continuous Attributes in C4.5", Journal of Artificial Intelligence Research, vol. 4, pp. 77-90, 1996.

[2] Harvinder Chauhan, Anu Chauhan, "Implementation of decision tree algorithm C4.5", International Journal of Scientific Research and Publications, vol. 3, pp. 1-3, 2013.

[3] Kalpesh Adhatrao, Aditya Gaykar, Amiraj Dhawan, Rohit Jha and Vipul Honrao, "Predicting Students Performance Using ID3 and C4.5 Classification Algorithms", International Journal of Data Mining & Knowledge Management Process, vol. 3, pp. 39-52, 2013.

[4] S. Moro, R. Laureano and P. Cortez. "Using Data Mining for Bank Direct Marketing: An Application of the CRISP-DM Methodology", Proceedings of the European Simulation and Modelling Conference - ESM'2011, pp. 117-121, 2011.

[5] Fabrizio Angiulli, Stefano Basta, Stefano Lodi and Claudio Sartori, "Distributed Strategies for Mining Outliers in Large Data Sets", IEEE Transactions on Knowledge and Data Engineering, vol. 25, pp. 1520-1532, 2013.

[6] Yakun Hu, Dapeng Wu and Antonio Nucci, "Fuzzy-Clustering-Based Decision Tree Approach for Large Population Speaker Identification", IEEE Transactions on Audio, Speech and Language Processing, vol. 21, pp. 762-774, 2013.

[7] Thomas Verbraken, Wouter Verbeke and Bart Baesens, "A Novel Profit Maximizing Metric for Measuring Classification Performance of Customer Churn Prediction Models", IEEE Transactions on Knowledge and Data Engineering, vol. 25, pp. 961-973, 2013.