A Novel Categorical Data Attribute Split Technique in Decision Tree Learning

D. Mabuni

***Abstract**: A new technique is proposed for splitting categorical data during the process of decision tree learning. This technique is based on the class probability representations and manipulations of the class labels corresponding to the distinct values of categorical attributes. For each categorical attribute aggregate similarity in terms of class probabilities is computed and then based on the highest aggregated similarity measure the best attribute is selected and then the data in the current node of the decision tree is divided into the number of sub sets equal to the number of distinct values of the best categorical split attribute. Many experiments are conducted using this proposed method and the results have shown that the proposed technique is better than many other competitive methods in terms of efficiency, ease of use, understanding, and output results and it will be useful in many modern applications.*

Keywords : Aggregated similarity measure and class probability representations, categorical split attribute, decision tree learning, splitting categorical data.

I. INTRODUCTION

Learning tools are becoming popular and common in effective data management and decision making applications. Decision tree is a famous data classification learning tool in a wide variety of real time tasks. It follows top down approach in greedy manner during its induction and exhibits highly pleasing properties including scalability, interpretability, fast convergent in data classification, divide and conquer, standard benchmark capability, robust, and with logarithmic time complexity of its operations. The algorithm C4.5 is the best decision tree algorithm that has been occupied in the best dominant position of the entire data mining algorithms set. Decision tree generation process is top-down process. Decision tree node splitting is the most important step in decision tree learning and how to select the best split attribute is heart of the decision tree creation tasks. It is a well known fact that larger decision trees produce less generalization performance results. By employing the best attribute selection techniques and the best stopping criteria it is possible to generate smaller and more general decision tree classifiers. Pruning is also one way of reducing the size of decision tree because decision tree classification increases as the size of the tree decreases.

The intent of this study is not to provide all the details of split measures available in the literature instead to explain in detail the proposed split measure for generation of efficient decision tree classifier consisting of smaller number of rules. Usually smaller decision trees produce small set of decision rules and represent more general decision features.

The normal process of decision tree generation is test all the attributes and evaluate the results one by one and then select the best attribute which gives the optimal measure. The splitting criteria is directly dependent in the purity measure value of the node and again the purity measure value of the node is dependent on the subsets of the distinct values of the attribute. In the present paper a new and novel categorical data splitting technique is proposed and an algorithm is developed, implemented, and then tested for the same algorithm.

Split attribute techniques are all formula based techniques and different split methods are available for node splitting. Proposed technique is one such formula based method for categorical attribute splitting and it is completely based on class probabilities in the subsets of individual distinct values of attributes.

$$Attribute\ split\ measure = \sum_{k=1}^{n} Abs(p_1 - p_2) \qquad (1)$$

Where
n is equal to the distinct categorical values of the split attribute and
p_1 is positive class probability and
p_2 is the negative class probability

For example, if the attribute A has 3 distinct values then the value of n = 3 and in simple notation the aggregate attribute split measure is represented as

$$Similarity(A) = \sum_{k=1}^{3} Abs(p_1 - p_2) \qquad (2)$$

Similarity (A) is the aggregation measure of similarities of distinct partitions corresponding to distinct categorical values of the attribute, A. Note that the value of n may be different for each attribute. Using the measure shown in equation (2) is used during decision tree generation for selecting the best split attribute.

Similarity(Age) = Similarity(Age = “Child”) +
Similarity(Age = “Young”) +

Similarity(Age = “Old”)

Manuscript received on April 02, 2020.
Revised Manuscript received on April 15, 2020.
Manuscript published on May 30, 2020.
* Correspondence Author

D. Mabuni*, Department of Computer Science, Dravidian University, Kuppam, India. Email: mabuni.d@gmail.com

Similarity(Age) is called the aggregate similarity measure of the attribute Age because Age attribute has three distinct categorical values. Aggregate similarity measure is taken into consideration for selecting the best splitting attribute during decision tree creation.

If the training dataset contains n number of attributes then for each attribute aggregate similarity in terms of sub sets similarity is computed and maximum of all these aggregate similarities is to be selected for the best attribute determination.

Aggregate similarity = maximum (Attribute-1, Attribute-2, Attribute-3, … , Attribute-n).

II. LITERATURE SURVEY

Decision trees are the most valuable, easy interpretable and frequently useful tools in data mining in particular for data classification and for effective decision making. The most critical task in decision tree learning is finding the best split attribute on the fly dynamically. Training and testing are the two fundamental tasks in decision tree induction. A.M Mahmood et. al. [1] proposed two new decision tree algorithms in order to improve area under the curve score value over the C4.5 decision tree classifier. B. Chandra et. al. [2] proposed a new split attribute technique, which is based on distinct class labels in a partition, for decision tree creation. B.H. Jun et. al. [3] pointed out that variants of gain ratio called normalized gain split attribute measures are also proposed and all these variants are intended to overcome the problem of gain ratio when the denominator is zero or very small. C. Drummond and R. Holte [4] explained sensitivity features of decision tree splitting technique. J. KENT MARTIN [5] Have discussed biases details of various splitting and stopping measures and also discussed about how ordering of the split attributes affects the efficiency and effectiveness of the pruning techniques.

J. R. Quinlan [6] ID3 is the very simple decision tree construction algorithm and probably it is the first international classification algorithm developed by Quinlan in 1986 and it uses information gain technique as its data splitting technique. ID3 algorithm can handle only categorical attributes and it does not use any pruning techniques. J. R. Quinlan [7] Quinlan proposed another decision tree creation algorithm which can handle both categorical and continuous attributes an it also uses pruning techniques to increase the generalization capabilities of the decision tree classifier. J. R. Quinlan [8] proposed an improved discretization of continuous attributes in decision tree creation.

L. Breiman et. al. [9] invented a very popular and the most important node data splitting measure called Gini index for decision tree creation. Also one must note that gain ratio and average gain ratio are the most important data splitting techniques. L. Jiang and C. Li [10] pointed out that information gain, minimum description length and class probabilities are the most important techniques used in finding the best split attribute during decision tree creation. R. Lopez De Mantaras [11] proposed a new split attribute measure for decision tree learning and it is based on distance between partitions. Sebastian Nowozin [12] pointed out that many split measures are biased and they must be replaced with improved estimators. Xinmeng Zhang and Shengyi Jiang, [13] proposed a new data splitting technique that uses cluster similarity measure for decision tree learning and experimentally verified that its performance is superior than many decision tree classifiers.

III. PROBLEM DEFINITION

Finding the best split attribute among the many candidate splits is the complex task in decision tree creation. Researchers are constantly proposing various types of split attribute techniques including categorical splits as well as numerical split attribute techniques by considering attributes in the training datasets. All the split attribute techniques are not the same and each one is inherently associated with certain pros and cons of its usage in certain applications.

The main problem is how to find the best split attribute technique with more and more generalized capability features and at the same time it must be portable across ample array of different domains of applications.

IV. PROPOSED CATEGORICAL SPLIT ATTRIBUTE TECHNIQUE IN DECISION TREE LEARNING

A new technique is proposed for splitting categorical data in decision tree construction. It is a probabilistic based approach and uses probabilities of classes of each distinct value of the categorical attribute. For example, if a particular attribute in the training dataset has five distinct categorical values, then five groups of class probabilities are to be computed and then these five types of probabilities are used in the special formula in order to calculate a single uniform similarity measure for each attribute of the training dataset. The data group similarity measure may be either maximum or minimum depending upon the problem context and attribute values of the training dataset.

In the given training dataset similarity is measured for each attribute separately and within each attribute similarity is measured separately for each categorical value of the attribute. For example in the given training dataset Age attribute has three distinct categorical values (Child, Young, and Old). Therefore, similarity (Age) is sum of similarities of similarity (Age = “Child”), similarity (Age = “Young”), and similarity (Age = “Old”).

Similarity (Age) = similarity (Age1) + similarity (Age2) + similarity (Age3). Similarity measures are computed for each attribute and then the attribute whose similarity is maximum is selected as the best split attribute and the data is divided based on the categorical values of the best split attribute.

Class 1 count of (Age = “Child”) = 8
Class 0 count of (Age = “Child”) = 4
Sum of class 1 and class 0 = 8 + 4 = 12
Maximum of (class 1 count, class 0 count) = 8 and its probability = 8/12
Minimum of (class 1 count, class 0 count) = 4 and its probability = 4/12
Absolute difference of class probabilities = 8/12 – 4/12 = (8 – 4)/12 = 4/12 = 0.3333.
Class 1 count of (Age = “Young”)
= 8

Class 0 count of (Age = “Young”) = 4
Sum of class 1 and class 0 = 8 + 4 = 12
Maximum of (class 1 count, class 0 count) = 8 and its probability = 8/12
Minimum of (class 1 count, class 0 count) = 4 and its probability = 4/12
Absolute difference of class probabilities = 8/12 – 4/12 = (8 – 4)/12 = 4/12 = 0.3333.
Class 1 count of (Age = “Old”) = 8
Class 0 count of (Age = “Old”) = 4
Sum of class 1 and class 0 = 8 + 4 = 12
Maximum of (class 1 count, class 0 count) = 8 and its probability = 8/12
Minimum of (class 1 count, class 0 count) = 4 and its probability = 4/12
Absolute difference of class probabilities = 8/12 – 4/12 = (8 – 4)/12 = 4/12 = 0.3333.
Similarity (Age) = 0.3333 + 0.3333 + 0.3333 = 1.0
Class 1 count of (BP = “High”) = 12
Class 0 count of (BP = “High”) = 0
Sum of class 1 and class 0 = 12 + 0 = 12
Maximum of (class 1 count, class 0 count) = 12 and its probability = 12/12 = 1.0
Minimum of (class 1 count, class 0 count) = 0 and its probability = 0/12 = 0
Absolute difference of class probabilities = 12/12 – 0/12 = (12 – 0)/12 = 12/12 = 1.0.
Class 1 count of (BP = “Low”) = 6
Class 0 count of (BP = “Low”) = 6
Sum of class 1 and class 0 = 6 + 6 = 12
Maximum of (class 1 count, class 0 count) = 6 and its probability = 6/12
Minimum of (class 1 count, class 0 count) = 6 and its probability = 6/12
Absolute difference of class probabilities = 6/12 – 6/12 = (6 – 6)/12 = 0/12 = 0.0.
Class 1 count of (BP = “Normal”) = 6
Class 0 count of (BP = “Normal”) = 6
Sum of class 1 and class 0 = 6 + 6 = 12
Maximum of (class 1 count, class 0 count) = 6 and its probability = 6/12
Minimum of (class 1 count, class 0 count) = 6 and its probability = 6/12
Absolute difference of class probabilities = 6/12 – 6/12 = (6 – 6)/12 = 0/12 = 0.0.
Similarity (BP) = 1.0 + 0.0 + 0.0 = 1.0
Class 1 count of (Sugar = “Yes”) = 18
Class 0 count of (Sugar = “Yes”) = 0
Sum of class 1 and class 0 = 18 + 0 = 18
Maximum of (class 1 count, class 0 count) = 18 and its probability = 18/18 = 1.0
Minimum of (class 1 count, class 0 count) = 0 and its probability = 0/18 = 0
Absolute difference of class probabilities = 18/18 – 0/18 = (18 – 0)/18 = 18/18 = 1.0.
Class 1 count of (Sugar = “No”) = 6
Class 0 count of (Sugar = “No”) = 12
Sum of class 1 and class 0 = 6 + 12 = 18
Maximum of (class 1 count, class 0 count) = 12 and its probability = 12/18
Minimum of (class 1 count, class 0 count) = 6 and its probability = 6/18
Absolute difference of class probabilities = 12/18 – 6/18 = (12 – 6)/18 = 6/18 = 0.3333
Similarity (Sugar) = 1.0 + 0.3333 = 1.3333
Also, similarity (Gender) = 0.5 + 0.5 = 1.0

TABLE-1 A sample patient dataset

Age	BP	Sugar	Gender	Happy
Child	Normal	Yes	M	1
Child	Normal	Yes	F	1
Child	Normal	No	M	0
Child	Normal	No	F	0
Child	High	Yes	M	1
Child	High	Yes	F	1
Child	High	No	M	1
Child	High	No	F	1
Child	Low	Yes	M	1
Child	Low	Yes	F	1
Child	Low	No	M	0
Child	Low	No	F	0
Young	Normal	Yes	M	1
Young	Normal	Yes	F	1
Young	Normal	No	M	0
Young	Normal	No	F	0
Young	High	Yes	M	1
Young	High	Yes	F	1
Young	High	No	M	1
Young	High	No	F	1
Young	Low	Yes	M	1
Young	Low	Yes	F	1
Young	Low	No	M	0
Young	Low	No	F	0
Old	Normal	Yes	M	1
Old	Normal	Yes	F	1
Old	Normal	No	M	0
Old	Normal	No	F	0
Old	High	Yes	M	1
Old	High	Yes	F	1
Old	High	No	M	1
Old	High	No	F	1
Old	Low	Yes	M	1
Old	Low	Yes	F	1
Old	Low	No	M	0
Old	Low	No	F	0

A hypothetical sample dataset is shown in Table-1. The dataset contains four predictor attributes and one class attribute. All attributes are categorical attributes. Age attribute has three distinct values, BP attribute has three distinct values, Sugar attribute has two distinct values and Gender attribute has two distinct values. Happy is the class label. The class label “0” indicates happy and “1” indicates not happy. The dataset is created based on the assumptions listed below:

1) If (Sugar = “Yes”) Then happy = 1
2) If (BP = “High”) Then happy = 1

If the person has sugar problem then the person is not happy. If the person is free from sugar but if the BP of the person is high then the person is not happy.

Similarity of Sugar is the maximum among all the attributes and it is equal to 1.3333. Hence, the best split attribute is “Sugar”. Data are partitioned into sub partitions using distinct values of Sugar attribute.

Same process is repeated with the remaining attributes in the successive levels of the decision tree creation.

In a similar manner similarities are computed and tabulated in respective tables. The proposed method is very simple to understand, develop, use, and apply in any desired application involving categorical attributes in the datasets. The method works for both two class and multi class datasets.

TABLE-2 Similarity of Age attribute

Attribute	Class	Class	Max	min	Difference
Age	1	0			probability
Child	8	4	8	4	4/12
Young	8	4	8	4	4/12
Old	8	4	8	4	4/12

Similarity (Age) = 4/12 + 4/12 + 4/12 = 12/12 = 1.0

TABLE-3 Similarity of BP attribute

Attribute	Class	Class	Max	min	Difference
BP	1	0			probability
High	12	0	12	0	12/12 = 1
Low	6	6	6	6	0
Normal	6	6	6	6	0

Similarity (BP) = 1 + 0 + 0 = 1.0

TABLE-4 Similarity of Sugar attribute

Attribute	Class	Class	Max	min	Difference
Sugar	1	0			probability
Yes	18	0	18	0	18/18 = 1
No	6	12	12	6	6/18 = 0.33

Similarity (Sugar) = 1 + 0.33 = 1.33

TABLE-5 Similarity of Gender attribute

Attribute	Class	Class	Max	min	Difference
Gender	1	0			probability
M	12	6	12	6	6/12 = 0.5
F	12	6	12	6	6/12 = 0.5

Similarity (Gender) = 0.5 + 0.5 = 1.0

Maximum similarity of Age, BP, Sugar and Gender = Maximum (1.0, 1.0, 1.33, 1.0) = 1.33, which corresponds to the Sugar attribute. Highest similarity attribute is Sugar, so, the dataset is divided into sub sets based on the distinct values of Sugar attribute and the sub sets are shown in respective tables.

TABLE-6 Sugar = “Yes” sub dataset

Age	BP	Gender	Happy
Child	Normal	M	1
Child	Normal	F	1
Child	High	M	1
Child	High	F	1
Child	Low	M	1
Child	Low	F	1
Young	Normal	M	1
Young	Normal	F	1
Young	High	M	1
Young	High	F	1
Young	Low	M	1
Young	Low	F	1
Old	Normal	M	1
Old	Normal	F	1
Old	High	M	1
Old	High	F	1
Old	Low	M	1
Old	Low	F	1

Sugar = “Yes” subset is 100 percent pure because all of its class labels (1) are same or homogeneous. That is its impurity is 0. So, there is no need to further sub divide this partition. The second partition of Sugar = “No” is shown in TABLE-7 and it contains both positive and negative instances and hence it must be sub divided into smaller groups of instances. Again the same process of finding aggregate similarity for each attribute is used to determine the best split attribute.

TABLE-7 Sugar = “No” partition sub dataset

Age	BP	Gender	Happy
Child	Normal	M	0
Child	Normal	F	0
Child	High	M	1
Child	High	F	1
Child	Low	M	0
Child	Low	F	0
Young	Normal	M	0
Young	Normal	F	0
Young	High	M	1
Young	High	F	1
Young	Low	M	0
Young	Low	F	0
Old	Normal	M	0
Old	Normal	F	0
Old	High	M	1
Old	High	F	1
Old	Low	M	0
Old	Low	F	0

TABLE-8 Age similarity after removing Sugar from the original dataset

Attribute	Class	Class	max	min	Difference
Age	1	0			probability
Child	2	4	4	2	2/6 = 0.33
Young	2	4	4	2	2/6 = 0.33
Old	2	4	4	2	2/6 = 0.33

Similarity (Age) = 0.33 + 0.33 + 0.3 = 1.0

TABLE-9 BP similarity after removing Sugar from the original dataset

Attribute	Class	Class	Max	min	Difference
BP	1	0			probability
High	6	0	6	0	6/6 = 1.0
Low	0	6	6	0	6/6 = 1.0
Normal	0	6	6	0	6/6 = 1.0

Similarity (BP) = 1.0 +1. 0 +1. 0 = 3.0

TABLE-10 Gender similarity after removing Sugar from the original dataset

Attribute Gender	Class 1	Class 0	Max	min	Difference probability
M	3	6	6	3	3/6 = 0.5
F	3	6	6	3	3/6 = 0.5

Similarity (Gender) = 0.5 + 0.5 = 1.0

Maximum similarity of Age, BP and Gender = Maximum (1.0, 3.0, 1.0) = 3, which corresponds to the BP attribute. Highest similarity attribute is BP, so, the dataset is divided into sub sets based on the distinct values of BP attribute and the sub sets are shown in respective tables. Here, BP = "High" means class label is 1; otherwise class label is 0.

TABLE-11 Sugar = "No" and BP = "High" Dataset

Age	Gender	Happy
Child	M	1
Child	F	1
Young	M	1
Young	F	1
Old	M	1
Old	F	1

TABLE-12 Sugar = "No" and (BP = "Low" or BP = "Normal") Dataset

Age	BP	Gender	Happy
Child	Normal	M	0
Child	Normal	F	0
Child	Low	M	0
Child	Low	F	0
Young	Normal	M	0
Young	Normal	F	0
Young	Low	M	0
Young	Low	F	0
Old	Normal	M	0
Old	Normal	F	0
Old	Low	M	0
Old	Low	F	0

V. ALGORITHM

```
Algorithm Categorical-Split-Decision-Tree(T, S)
Input:
    S is the set of training dataset with categorical attributes
    T is a pointer to root node
Output:
    Decision tree classifier

1.if(T.size < threshold or attribute set is empty)
2.    create a leaf node
3.    return
4.end-if
5. A = find the set of distinct categorical attributes in the dataset S
6.for each attribute $A_i$ in the set A find distinct values of $A_i$
7.for each attribute $A_i$ in A do
8.    for each value $v_k$ in $A_i$ do
9.        find group $Gv_k$
10.       find the similarity of the group $Gv_k$1
11.   end-for k
12.end-for i
13.for each attribute $A_i$ find aggregated similarities
14.find the best attribute $A_k$ whose aggregated similarity is maximum
15.for each distinct value of the best attribute $A_k$ do
16.    $T_k$ = create new node
17.    call Categorical-Split-Decision-Tree($T_k$, $S_k$)
18.end-for
19.print decision tree
```

A. Algorithm Explanation

Lines-1, 2, 3 and 4 belongs to a base case and the current node is converted into a leaf node when node size is less than a specified threshold or attributes set is empty

Line-5: finds the set of distinct categorical attributes of the training dataset.

Line-6: finds distinct categorical values of each attribute A_i in the given training dataset

Lines-7, 8, 9, 10, 11 and 12: for each distinct value A_k of each attribute A_i groups of values and class labels are created and then similarities for groups are computed

Line-13: attribute wise aggregate similarities are computed

Line-14: finds the best split categorical attribute corresponding to maximum aggregated similarity value

Lines-15, 16, 17 and 18 child nodes are created recursively for each distinct value of the best attribute and the process is repeated until all the input attributes are exhausted or node size falls below a specified threshold.

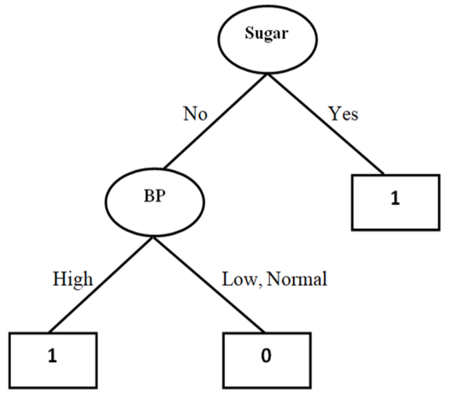

Fig-1 Decision Tree for the dataset shown in TABLE-1

If(Sugar = 1) then Happy = 1
If(sugar = 0 and if(BP) = High then Happy = 1
If(sugar = 0 and if(BP) = (Low or Normal) then Happy = 0

VI. EXPERIMENTS

Experiments are conducted by taking both simulated and standard UCI machine learning datasets. After thorough investigation of the experimental results it has been observed that the proposed categorical data split technique is reliable and will become a standard benchmark method in the future for splitting the categorical attributes in the decision tree learning.

Retrieval Number: A2568059120/2020©BEIESP
DOI:10.35940/ijrte.A2568.059120
Journal Website: www.ijrte.org

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication

Experiments are conducted on both two class and three class datasets. All the persons without sugar and BP are happy. This is true in majority of the real life situations also. After processing the dataset by the proposed categorical split attribute algorithm the output results resembles the imposed rules on the dataset. That is results are perfectly matched with the pre assumed conditions applied on the dataset. This shows the correctness of the proposed technique.

From the experiments it is observed that gender is independent of happy class label. That is happiness is not dependent on the gender feature. Also note that in the data happiness is not influenced by age feature. The resulted tree is simple, compact and represents a small set of rules which are easy interpretable for all people.

TABLE-13 Experiment Results

Dataset Name	Training dataset size, attributes number	Height of the tree	Number of leaves	Accuracy with the same training dataset
All electronics	14,5	3	5	100.0
Patient	36,5	3	3	100.0
Balloon-1	16,5	3	3	100.0
Balloon-2	16,5	5	10	100.0
Balloon-3	16,5	5	10	100.0
Balance Scale	625,5	5	429	100.0
SPECT	81,23	19	36	100.0
Nursery	12960,9	9	2062	100.0

Pruning techniques are useful for reducing the size of the decision tree and increasing the capability of generalizing ability of the tree. In this study pre pruning techniques are applied.

VII. CONCLUSION

Though data splitting in the node is a difficult task but it is compulsory in decision tree induction. Proposed categorical data split technique is convenient, efficient and effective for decision tree classifier creation. In the future the best efforts will be applied for finding optimal split attribute techniques involving both continuous and categorical attributes in the datasets. One split attribute technique may not be perfectly suitable for all applications. Also different new methods will be searched and then investigated to know how the same split attribute technique must be mould appropriately for the suitability of the selected application in real time situations. In the future more number of tests will be conducted by using separate test datasets.

REFERENCES

1. A. M. Mahmood, K. M. Rao, K. K. Reddi, et al. A Novel Algorithm for Scaling up the Accuracy of Decision Trees. International Journal on Computer Science and Engineering, vol.2, pp. 126-131, 2010.
2. B. Chandra, RaviKothari, and PallathPaul, "A new node splitting measure for decision tree construction", ELSEVIER, Pattern Recognition 43 (2010) 2725–2731, Pattern Recognition,
3. B.H. Jun, C.S. Kim, J. Kim, A new criterion in selection and discretization of attributes for the generation of decision trees, IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (12) (1997) 1371–1375.
4. C. Drummond, R. Holte. Exploiting the cost (in) sensitivity of decision tree splitting criteria. Proceedings of the Seventeenth International Conference on Machine Learning .pp. 239–246,2000.
5. J. KENT MARTIN, "An Exact Probability Metric for Decision Tree Splitting and Stopping", Machine Learning, 28, 257–291 (1997) 1997 Kluwer Academic Publishers. Manufactured in The Netherlands.
6. J. R. Quinlan, Induction of decision trees. Machine Learning, vol.1,pp. 81-106, 1986.
7. J. R. Quinlan, C4.5: Programs for machine learning. 1st ed.San Mateo, CA: Morgan Kaufmann, 1993.
8. J. R. Quinlan, Improved use of continuous attributes in C4.5, Journal of Articial Intelligence Research, vol.4, pp.77-90, 1996.
9. L. Breiman, J. Friedman, R. Olsen, C. Stone, Classification and Regression Trees, Wadsworth International, 1984.
10. L. Jiang, C. Li, An Empirical Study on Class Probability Estimates in Decision Tree Learning, journal of software,vol.6, pp.1368-1372, 2011.
11. R. Lopez De Mantaras, "A distance based attribute selection measure for decision tree induction", Machine Learning, 6, 81-92 (1991) © 1991 Kluwer Academic Publishers, Boston. Manufactured in The Netherlands.
12. Sebastian Nowozin, "Improved Information Gain Estimates for Decision Tree Induction", Appearing in Proceedings of the 29 th International Conference on Machine Learning, Edinburgh, Scotland, UK, 2012. Copyright 2012 by the author(s)/owner(s).
13. Xinmeng Zhang and Shengyi Jiang, "A Splitting Criteria Based on Similarity in Decision Tree Learning", JOURNAL OF SOFTWARE, VOL. 7, NO. 8, AUGUST 2012, © 2012 ACADEMY PUBLISHER doi:10.4304/jsw.7.8.1775-1782

AUTHORS PROFILE

D. Mabuni, completed M.Sc. (Computer Science), MCA and M.Phil. (Computer Science). Currently working as Assistant Professor in the Department of Computer Science at Dravidian University, Kuppam, Andhra Pradesh, India. My interested research areas are Data Mining, Databases, and User Interfaces.