DEVELOPMENT OF A COMPACT LINGUISTIC RULES-TREE (CLR-Tree): THE FIRST PHASE

[1]Rashid Hafeez Khokhar, [2] Mohd Noor Md Sap

Faculty of Computer Science and Information System
University Technology of Malaysia, K.B. 791
81310 Skudai, Johor, Malaysia
Tel: (607)-5532419, Fax: (607) 5565044
[1]rashid@siswa.utm.my, [2] mohdnoor@fsksm.utm.my

Abstract :
Classification in data mining is very extensive research area. Decision trees have been found very effective for classification of huge and frequently modifiable databases e.g., Stock Exchange, Shopping Mall etc. We build a decision tree from a training set consists of two phases. In the first phase the initial Linguistic Rules-Tree (LR-Tree) has been constructed. In LR-Tree we have combined fuzzy logics and decision tree. First we evaluate fuzzy membership function from training data for each attribute in class then apply our fuzzy linguistic approach which is associated with decision tree that provides for a fine grain description of classified items adequate for human reasoning. Consequently our approach will be able to handle training data with missing attribute values, handling attributes with differing costs, improving computational efficiency. But LR-Tree may not be the best generalization due to over-fitting so in the second phase, we will propose a novel frequent pattern mining tree called Compact Linguistic Rules-Tree (CLR-Tree) that remove some branches and nodes to improve the accuracy of the classifier. In this paper, we have concentrated on construction phase and hope that after completing the construction phase we will proof that the CLR-Tree is efficient and scalable for mining both long and short frequent patterns.

Keywords: Data mining, Classification, Decision tree, Fuzzy logics, Linguistic Approach, Frequent Pattern Mining.

1. Introduction

Classification of data is one of the important tasks in data mining. Various methods for classification have been proposed. Apriori Approach, Tree-Projection ID3, and C4.5 are popular approaches. *Apriori Approach* [1] is a well known technique for large itemsets generation. In the Apriori Approach a level wise algorithm is used in order to generate itemsets. The idea is to generate candidates for (k+1)-itemsets from k-itemsets by using joins over k-itemsets. The candidates for (k+1)-itemsets are then validated by a scan over the database. An Apriori algorithm may still suffer from the following two costs. 1) It is costly to handle a huge number of candidates' sets. 2) It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns.

The *Tree-Projection* algorithm proposed by Agarwal et al. [2] recently is an interesting algorithm, which constructs a lexicographical tree and projects a large database into a set of reduced, item-based subdatabases based on the frequent patterns mined so far. Tree-Projection may still encounter difficulties at computing matrices when the database is huge. The study in Tree-Projection [2] has developed some smart memory caching methods to overcome this problem. However, it could be wise not to generate such huge matrices at all instead of finding some smart caching techniques to reduce the cost [3]. Moreover, even if the matrix can be cached efficiently, its computation still involves some nontrivial overhead. And also when there are many long transactions containing numerous frequent items, transaction projection become a nontrivial cost of Tree-Projection.

ID3 is another popular classification algorithm proposed by Quinlan [4, 5] in 1979, which makes a decision tree for classification from symbolic data but ID3 is generally not suitable in cases where numerical values are to be operated upon. Since most real life problems deal with non-symbolic (numeric, continuous) data, they must be discretized prior to attribute selection. Another problem with ID3 is that it cannot provide any information about the intersection region where the pattern classes are overlapping.

In data mining important factors are efficiency and comprehensibility. The discovered knowledge should well describe the characteristics of the data and they should be easy to understand in order to facilitate better understanding of data. In the first phase, we propose an efficient Linguistic Rules-Tree (LR-Tree) which is the combination of Fuzzy Linguistic Approach and C4.5 [6]. C4.5 is an extended form of ID3 [4, 5] introduced by Quinlan 1993 [6] for inducing Classification Models, from huge databases. C4.5 uses a windowing technique which works as follows. A small sample is drawn from the dataset to build an initial tree. This sample is augmented with records that were misclassified in the initial tree. This process is repeated for a number of iterations. In LR-Tree we will describe the following issues: handling training data with *missing attribute values*, handling attributes with differing costs, improving computational efficiency and provides for a fine grain description of classified items adequate for human reasoning.

In second phase, we will present our Compact Linguistic Rules-Tree (CLR-Tree). In that phase, we will use frequent pattern mining method. Frequent pattern mining [7, 8] plays an essential role in mining associations [9-20], correlations [21, 22], sequential patterns [23, 24], episodes [25], multi-dimensional patterns [26], max-patterns [27], partial periodicity [28], emerging patterns [29], and many other important data mining tasks. In second phase, we will reconstruct our LR-Tree and introduce new CLR-Tree, which is extended form of LR-Tree structure storing crucial, quantitative information about frequent pattern.

The paper is organized as follows: in the next section we briefly overview the components of Linguistic Rules-Tree (LR-Tree), in section 3 we will construct LR-Tree. Finally in section 4 concluding remarks and future works are presented.

2. Components of Linguistic Rules-Tree (LR-Tree)

Linguistic Rules-tree (LR-Tree) is an alternative form of C4.5 [30] with some extra features as: handling training data with missing attribute values, handling attributes with differing costs, improving computational efficiency and provides for a fine grained description of classified items adequate for human reasoning. Using linguistic terms to represent the revealed regularities and exceptions, this approach is especially useful when the discovered rules are presented to human experts for examination because of the affinity with the human knowledge representation. The use of fuzzy technique allows the prediction of attribute values to be associated with degree of membership. Our approach is therefore, able to deal with the cases that an object can belongs to more than one class. And also our approach is more resilient to noise and missing data values because the use of fuzzy technique.

LR-Tree is capable of mining fuzzy rules in large databases without any need for user specified thresholds or mapping of quantitative into binary attributes. A fuzzy rule describes an interesting relationship with two or more linguistic terms. After describing interesting rules we will associate these rules with decision tree at the same level, detail explanation about LR-Tree will be depicted in section 3.

Linguistic Rules-Tree (LR-Tree) is a three steps procedure. In the first step in figure 1 pre-classification: first we arrange all set of attributes in ascending order. Then evaluate maximum and minimum values for every set of attributes in each class. After finding appropriate intervals we assign all set of attributes of each class to artificial values with in the range [0, 1]. By converting all set of attributes in to artificial values, we will be able to define linguistic variables more precisely.

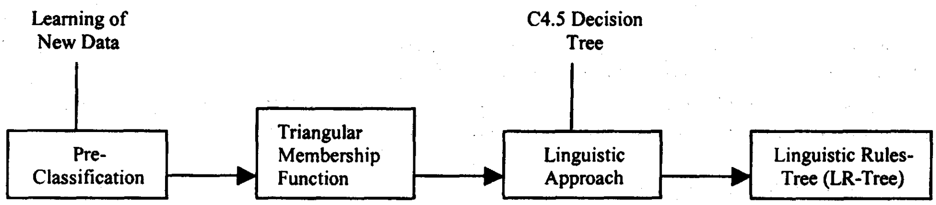


Figure 1: The Components of LR-Tree Construction

Membership functions are very important in fuzzy rules. Membership functions affect not only the performance of fuzzy rule based system but also the comprehensibility of fuzzy rules. In second step we adopt triangular fuzzy membership functions based on three numbers, actually these variables are related to linguistic variables. In third step LR-tree is composed of nodes, terminal nodes and arcs. A node represents an attribute to partition the pattern space. A terminal node is associated with linguistic variables. An arc is associated with a fuzzy set of the attribute correspond to the parent node. LR-tree is the combination of fuzzy linguistic approach and C4.5 [30]. LR-tree is defined as each path from the root node to a terminal node corresponds to a fuzzy rule and it also corresponds to a partitioned fuzzy subspace of the whole pattern space. We will elaborate these components of LR-Tree in more detail in section 2.1 to 2.4.

2.1 Pre-Classification

Let $\max(a_i)$ and $\min(a_i)$ be the maximum and minimum functions for set of attributes in each class respectively. Suppose the function for evaluating the difference between maximum and minimum values in each class denoted by $\text{diff}(a_i)$ and defined as

$$\text{diff}(a_i) = \max(a_i) + \min(a_i) \quad \text{---------------(1)}$$

Let $\text{CInt}(a_i)$ be the class interval for set of attributes and define as,

$$\text{CInt}(a_i) = \text{diff}(a_i) / n \quad \text{--------------(2)}$$

Where n is an integer number for reasonable groups which will be taken by experts choice.
So with this class interval we can define every class just adding this interval in $\min(a_i)$ and find ten class which is not exceed to $\max(a_i)$.

Example 1 shows how we arrange raw data in different number of groups, which has been taken from Stock Exchange databases. After doing this example we will see that how our system learn new data.

Example 1: Stock Exchange Indices trading hours: 1st session: 09:00 to 12:30, 2nd session: 14:30 to 1700

Table 1: Trading during 1st and 2nd session in Stock Exchange.

INDICES	SALE	PURCHASE	TURNOVER
MESDAQ	79.56	85.63	-6.07
SYARIAH	99.05	88.65	10.4
INDUSTRIAL	1384.68	1463.89	-79.21
PLANTATION	1834.93	1834.93	0
TECHNOLOGY	37.87	45.23	-7.36
CONSTRUCTION	155.55	165.52	-9.97
PROPERTY	529.33	523.47	5.86
EMAS	157.38	153.35	4.03
IND-PROD	64.88	69.45	-4.57

The above table shows the share transaction of different companies in Kuala Lumpur Stock Exchange during two sessions. First column shows different indices and the next columns are sale, purchase and turnover of every company.

First we arrange a set of attributes (e.g., sale, purchase, and turnover), then find maximum and minimum values in each attribute. Suppose $\max(a_{is})$ s $\in$ sale, $\max(a_{ip})$ m p $\in$ purchase, $\max(a_{it})$ t $\in$ turnover are the maximum functions of sale (s), purchase (p) and turnover (t). We just show here only for sale attribute.
Maximum and Minimum values for sale (s) have been taken from the table 1,

$\max(a_{is}) = 1834.93$, $\min(a_{is}) = 37.87$

After getting maximum and minimum values in each attribute we will find out the common differences with help of equation 1. Let $\text{diff}(a_{is})$, $\text{diff}(a_{ip})$, $\text{diff}(a_{it})$ are the difference functions that can be used to define difference between maximum and minimum value in set of attributes, these differences can be defined as:

$\text{diff}(a_{is}) = [\max(a_{is}) - \min(a_{is})]$

$diff(a_{is}) = 1834.93 - 37.87 = 1797.06$

Suppose class intervals for each attribute denoted by $CInt(a_{is})$, $CInt(a_{ip})$, $CInt(a_{it})$ and define as:

$CInt(a_{is}) = diff(a_{is}) / n$

Where n is number of groups, it depends on system experts choice. Here we take n as 10.

$CInt(a_{is}) = 1797.06 / 10 = 179.706$

Then minus and plus the above class-intervals for every records in sale. Similarly do the same job for purchase and turnover. Finally we get the Table 2 which shows the results of Pre-Classification for sale, purchase and turnover.

Table 2: Pre-Classification of Crisp values

NO	CLASSIFICATION FOR SALE	CLASSIFICATION FOR PURCHASE	CLASSIFICATION FOR PURCHASE	ARTIFICIAL VALUES
1	-141.836 — 217.576	45.23 — 224.2	-79.21 — -70.249	0.1
2	217.576 — 397.282	224.2 — 403.17	-70.249 — -61.288	0.2
3	397.282 — 576.988	403.17 — 582.14	-61.288 — -52.327	0.3
4	576.988 — 756.694	582.14 — 761.11	-52.327 — -43.366	0.4
5	756.694 — 936.4	761.11 — 940.08	-43.366 — -34.405	0.5
6	936.4 — 1116.106	940.08 — 1119.05	-34.405 — -25.444	0.6
7	1116.106 — 1295.812	1119.05 — 1298.02	-25.444 — -16.483	0.7
8	1295.812 — 1475.518	1298.02 — 1476.99	-16.483 — -7.522	0.8
9	1475.518 — 1655.224	1476.99 — 1655.96	-7.522 — 1.439	0.9
10	1655.224 — 2014.63	1655.96 — 1834.93	1.439 — 10.4	1.0

We assign artificial values to every group as shown in table 2. So after pre-classification we will use artificial values as universe of discourse in fuzzy linguistic approach.

2.2 Triangular Fuzzy Membership Function

Membership functions are very important in fuzzy rules. Membership functions affect not only the performance of fuzzy rule based systems but also the comprehensibility of rules. In this paper we adopt triangular fuzzy numbers for membership functions and we generate membership function based on histogram analysis. Our justification to doing this is that more frequently observed attributes values (or value ranges) of data should be more significant than less frequently observed ones in classification.

To generate fuzzy membership functions, we first calculate the histogram from the training data for each attribute of each class and then we smooth it using the moving average method to remove shallow local minima and maxima points of the histogram. We normalized the smoothed histogram to adjust the height of the local maxima to the global maximum. Next we generate a fuzzy membership function corresponding to each maximum point of the histogram so that the center point of the triangular fuzzy membership function corresponds to the maximum point and the end points are determined by connecting the maximum point to its nearest local minimum points in both sides. In pattern classification problems it is important to identify regions of critical attribute values determined by minima and maxima points and this method ensures that the critical regions of the histogram are properly mapped into fuzzy membership functions. Figure 2 illustrates fuzzy membership function generation based on histogram analysis.

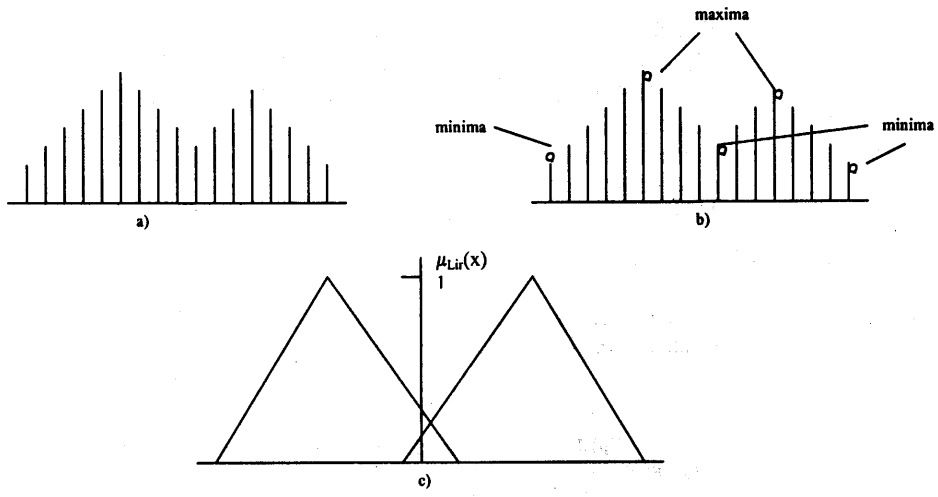


Figure 2: Fuzzy membership function based on histogram analysis

A triangular membership function can be represented by a triple numbers (l, m, h), where l is lower number, m is mid point and h is higher number in the triangular membership function.
The triangular membership function is denoted as $\mu_A(x)$ and is defined as:

$$\mu_A(x) = \begin{cases} \dfrac{x-l}{m-l} & : \text{If, } l \leq x \leq m \\ \dfrac{h-x}{h-m} & : \text{If, } m \leq x \leq h \\ 0 & : \text{Otherwise} \end{cases} \qquad \text{----------- (3)}$$

In the above membership first we find out the domain of input values and ensure that input values are in the range of our membership function. Membership function has three possible values of input variable x. as shown above in equation 3.

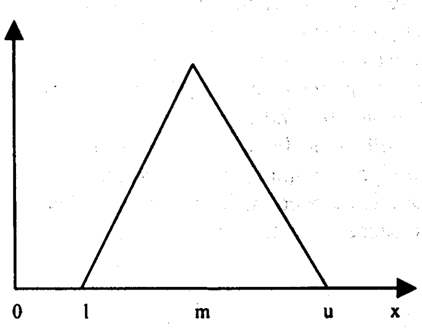


Figure 3: Triangular fuzzy membership function

2.3 Fuzzy Linguistic Approach

Given a set of records, D, each of which consist of set of attributes $A = [A_1,\ldots\ldots,A_n]$, where A_i, $(i = 1,\ldots,n)$ may be quantitative or categorical. For any tuple (record), $d \in D$, $d[A_i]$ denotes the value a_i in d for attribute A_i. For any quantitative attribute, $A_i \in A$, let $X = [l_i, u_i]$ subset of $\check{R}$ denote the area of universe of discourse. Suppose linguistic terms associated with some quantitative attribute, $Ai \in A$ as ℓ_{ir}, $r = 1,\ldots\ldots,sv$, so that corresponding fuzzy set, L_{ir} can be defined for each ℓ_{ir}.

Fuzzy set L_{ir} of universe X is define by function $\mu_{Lir}(x)$ called the membership function of set L_{ir}.

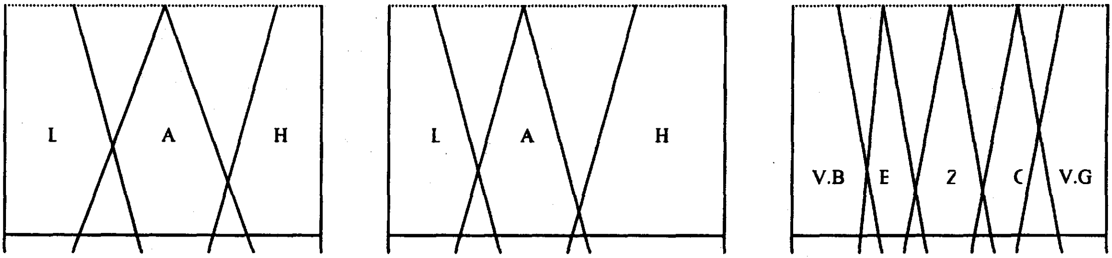
$$\mu_{Lir}(x) : X \rightarrow [0, 1].$$

Linguistic Approximation is mapping from the crisp sets C_i of all fuzzy sets L_{ir} of the universe of discourse X into the language $\zeta = L_{ir}$, set of linguistic variables.

Fuzzy sets can have a variety of shapes. However, a triangle or a trapezoid can often provide an adequate representation of the expert knowledge and at the same time significantly simplifies the process of computation. With the help of example 1, here we proceed for further step. Figure 4 is representing our fuzzy sets for linguistic variables s, p and t. Horizontal direction presents universe of discourse $X \in$ sale (s), purchase (p), and turnover (t). And vertical direction presents degree of membership [0, 1].

Since each linguistic term is represented by a fuzzy set, we have a set of fuzzy sets, $S = \{ \ell_{ir} \mid i = 1,\ldots n, r = 1,\ldots\ldots sv\}$. Given a record $d \in D$, and a linguistic term, $\ell_{ir} \in S$, which is in turn, represented by a fuzzy set, $L_{ir} \in S$, the degree of membership of the value in d with respect to ℓ_{ir} is given by $\mu_{Lir}(d[A_i])$. In other words, d is characterized by the term ℓ_{ir} to the degree $\mu_{Lir}(d[A_i])$. If $\mu_{Lir}(d[A_i]) = 1$, d is completely characterized by the term ℓ_{ir}. If $\mu_{Lir}(d[A_i]) = 0$, d is not characterized by the term ℓ_{ir} at all.

If $0 < \mu_{Lir}(d[A_i]) < 1$, d is partially characterized by the ℓ_{ir}.



a) Sale b) Purchase c) Turnover

Figure 4: Fuzzy Sets for Sale (s), Purchase (p) and Turnover (t)

2.4 Decision Tree

Decision tree is one of the most widely used classification techniques especially in artificial intelligence. Their popularity is basically due to their ability to express knowledge in a formalism that is often easier to interpret by experts and even by ordinary users. Systems based on this approach use an information theoretic measure of entropy for assessing the discriminatory power of each attribute. The most important feature of decision trees is their capability to break down a complex decision making process into a collection of simpler decisions and thereby, providing an easily interpretable solution [31].

A decision tree classifier recursively partitions the training set until each partition consists entirely, or almost entirely, of records from one class. Each non leaf node of the tree contains a split criterion which is a test on one or more attributes and determines how the data is partitioned.

Table 3: Training set

INDICES	SALE	PROFIT	TAX	APPROVAL
MESDAQ	400,000 RM	50,000 RM	2500 RM	No
SYARIAH	560,000 RM	70,000 RM	3500 RM	Yes
INDUSTRIAL	800,000 RM	100,000 RM	5000 RM	Yes
PLANTATION	960,000 RM	120,000 RM	6000 RM	Yes
EMAS	360,000 RM	85,000 RM	4250 RM	Yes
CONSTRUCTION	760,000 RM	95,000 RM	4000 RM	No
PROPERTY	208,000 RM	46,000 RM	2300 RM	No
TECHNOLOGY	896,000 RM	112,000 RM	5600 RM	Yes
IND-PROD	384,000 RM	48,000 RM	2400 RM	No

Figure 5 is a sample decision tree classifier based on the training set shown in Table 3. Three split criteria, (Sale > 100,000), (Profit > 50, 000) and (Tax 5%), partition the dataset into Declined and Approved classes. This classification model can then be used to process future loan applicants by classifying them into Declined or Approved classes according to the decision tree.

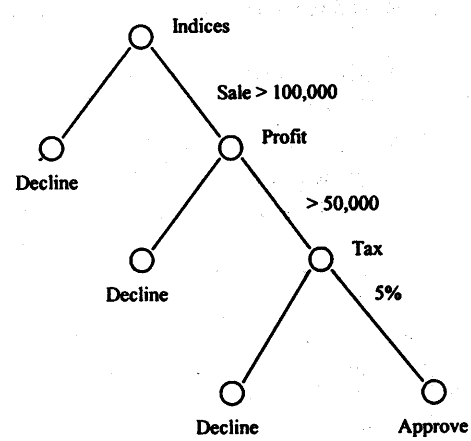


Figure 5: Decision Tree

3. Making of Linguistic Rules-Tree (LR-Tree)

A LR-Tree is composed of leaf nodes, terminal nodes and arcs. A leaf node represents an attribute to partition the pattern space. A terminal space node is associated with a linguistic variables and arc is associated with a fuzzy set of the attribute corresponding to the parent node. Figure 6 illustrates a typical form of LR-tree. In a LR-tree each path form the root node to a terminal node corresponds to a fuzzy subspace of the whole pattern space. In figure
6 dash-rectangles (terminal nodes) correspond to linguistic variables, leaf represent attributes, and arcs represent fuzzy sets.

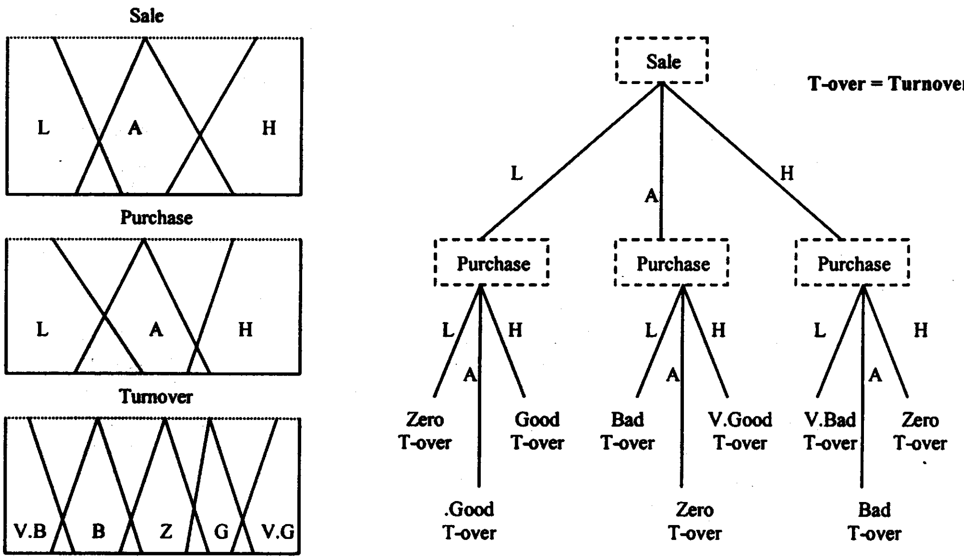


Figure 6: LR-Tree Construction for Stock Exchange Indices

Equation (4) calculates the membership degree of training data X_i on a fuzzy subspace partitioned by the fuzzy sets from the root node to node i.

$$V_{il} = \begin{cases} \min_{f \in C_i} (\mu_f(x_{fl})) & : C_i \neq \Phi \\ 1 & : C_i \neq \Phi \end{cases} \qquad \text{---------------- (4)}$$

In equation (4), $C_i = \{f_{i1}, f_{i2}, \ldots, f_{ik}\}$, a class of fuzzy sets each of which is associated with an arc on path P_i from the root node to node i and x_{fl} is an attribute value for membership function f of training data X_i. In our algorithm, V_{il} represents the membership degree that training data X_i belongs of the fuzzy subspace corresponding to path P_i and it is used to calculate the entropy that measure the degree that the fuzzy subspace contains a single class of patterns.

The entropy of attribute F for node i, E_F^i is defined as in equation (5), where C is the set of classes, D is the set of training data.

$$E_F^i = \sum_j (pijI^j) \text{---------------(5)}$$

$$I^j = -\sum_{k \in C} (p_k^j \log_2 p_k^j) \text{----------(6)}$$

$$p_k^j = \frac{\sum_{m \in K} V_{jm}}{\sum_{l \in D} V_{jl}} \text{--------------(7)}$$

$$p_{ij} = \frac{\sum_{l \in D} V_{jl}}{\sum_{l \in D} V_{il}} \text{--------------------(8)}$$

In the above equations, j is a child node of node i, corresponding to a fuzzy membership function for an attribute F. Equation (6) calculates the information gain for node j, equation (7) represents the probability that node i represents class k, and equation (8) represents the possibility compared to other sibling nodes.

4. Conclusion

To deal with decision tree and fuzzy rules, we present a new LR-tree, which employs linguistic terms to represent regularities and exceptions discovered. These linguistic terms can be defined by fuzzy sets so that based on their membership function we developed linguistic rules extracted from a decision tree. A novel concept for measuring the goodness of a decision tree, in terms of its compactness (size) and efficient performance is introduced. Linguistic rules are quantitatively evaluated using new indices. New fuzziness measures, in terms of class memberships, are used at the node level of the tree to take care of overlapping classes.

In LR-Tree, we will be able to handle training data with missing attribute values, handling attributes with differing costs, improving computational efficiency and provides for a fine grained description of classified items adequate for human reasoning. In our second phase, we will present Compact Linguistic Rules-Tree (CLR-Tree), which is extended form of LR-Tree. In that phase we will reduce the size of tree by using frequent pattern mining methods. LR-Tree will be extended due to following research issues: avoiding over-fitting the data (determining how deeply to grow a decision tree), reduced error pruning, rule post-pruning, handling continuous attributes (e.g., temperature), and choosing an appropriate attribute selection measure.

Reference

[1] S.M Weiss, Predictive Data Mining: A Practical Guide, San Francisco, CA: Morgan Kaufmann, 1998.

[2] R. Agarwal, C. Aggarwal, and V. V. V. Prasad. A tree projection algorithm for generation of frequent itemsets. In Journal of Parallel and Distributed Computing (Special Issue on High Performance Data Mining), (to appear), 2000.

[3] J. Han, J. Pei, and Y. Yin. Mining frequent patterns without candidate generation. In SIGMOD'00, pages 1--12.

[4] S. R. Safavian and D. Landgrebe, ``A survey of decision tree classifier methodology," IEEE Trans. Syst, Man, Cybern., vol. 21, pp. 660--674, 1991.

[5] P. E. Utgoff "Incremental induction of decision trees" Machine Learning, 4, pp 161-186, 1989.

[6] J. R. Quinlan "C4.5: Programs for machine learning" Morgan Kaufmann, San Mateo, Ca, 1993.

[7] R. Agrawal and R. Srikant. Fast algorithms for mining association rules. In Proc. 1994 Int. Conf. Very Large Data Bases, pages 487#499, Santiago, Chile, September 1994.

[8] M. Klemettinen, H. Mannila, P. Ronkainen, H. Toivonen, and A.I. Verkamo. Finding interesting rules from large sets of discovered association rules. In Proc. 3rd Int. Conf. Information and Knowledge Management, pages 401#408, Gaithersburg, Maryland, Nov. 1994.

[9] A. Savasere, E. Omiecinski, and S. Navathe. An e#cient algorithm for mining association rules in large databases. In Proc. 1995 Int. Conf. Very Large Data Bases, pages 432#443, Zurich, Switzerland, Sept. 1995.

[10] J.S. Park, M.S. Chen, and P.S. Yu. An e#ective hash-based algorithm for mining association rules. In Proc. 1995 ACM-SIGMOD Int. Conf. Management of Data, pages 175#186, San Jose, CA, May 1995.

[11] R. Srikant and R. Agrawal. Mining generalized association rules. In Proc. 1995 Int. Conf. Very Large Data Bases, pages 407#419, Zurich, Switzerland, Sept. 1995.

[12] J. Han and Y. Fu. Discovery of multiple-level association rules from large databases. In Proc. 1995 Int. Conf. Very Large Data Bases, pages 420#431, Zurich, Switzerland, Sept. 1995.

[13] R.J. Miller and Y. Yang. Association rules over interval data. In Proc. 1997 ACM-SIGMOD Int. Conf. Management of Data, pages 452#461, Tucson, Arizona, May 1997.

[14] B. Lent, A. Swami, and J. Widom. Clustering association rules. In Proc. 1997 Int. Conf. Data Engineering #ICDE'97#, pages 220#231, Birmingham, England, April 1997.

[15] S. Brin, R. Motwani, J. D. Ullman, and S. Tsur. Dynamic itemset counting and implication rules for market basket analysis. In Proc. 1997 ACM-SIGMOD Int. Conf. Management of Data, pages 255#264, Tucson, Arizona, May 1997.

[16] S. Sarawagi, S. Thomas, and R. Agrawal. Integrating association rule mining with relational database systems: Alternatives and implications. In Proc. 1998 ACM-SIGMOD Int. Conf. Management of Data, pages 343#354, Seattle, Washington, June 1998.

[17] R. Srikant, Q. Vu, and R. Agrawal. Mining association rules with item constraints. In Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining #KDD'97#, pages 67#73, Newport Beach, California, August 1997.

[18] R. J. Bayardo, R. Agrawal, and D. Gunopulos. Constraint-based rule mining on large, dense data sets. In Proc. 1999 Int. Conf. Data Engineering #ICDE'99#, Sydney, Australia, April 1999.

[19] R. Ng, L. V. S. Lakshmanan, J. Han, and A. Pang. Exploratory mining and pruning optimizations of constrained associations rules. In Proc. 1998 ACM-SIGMOD Int. Conf. Management of Data, pages 13#24, Seattle, Washington, June 1998.

[20] M. J. Zaki, S. Parthasarathy, M. Ogihara, and W. Li. Parallel algorithm for discovery of association rules. Data
Mining and Knowledge Discovery, 1:343#374, 1997.

[21] S. Brin, R. Motwani, and C. Silverstein. Beyond market basket: Generalizing association rules to correlations. In Proc. 1997 ACM-SIGMOD Int. Conf. Management of Data, pages 265#276, Tucson, Arizona, May 1997.

[22] C. Silverstein, S. Brin, R. Motwani, and J. Ullman. Scalable techniques for mining causal structures. In Proc. 1998 Int. Conf. Very Large Data Bases, pages 594#605, New York, NY, August 1998.

[23] R. Agrawal and R. Srikant. Mining sequential patterns. In Proc. 1995 Int. Conf. Data Engineering, pages 3#14,
Taipei, Taiwan, March 1995.

[24] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. In Proc.5th Int. Conf. Extending Database Technology #EDBT#, pages 3#17, Avignon, France, March 1996.

[25] H. Mannila, H Toivonen, and A. I. Verkamo. Discovery of frequent episodes in event sequences. Data Mining and Knowledge Discovery, 1:259#289, 1997.

[26] M. Kamber, J. Han, and J. Y. Chiang. Metarule-guided mining of multi-dimensional association rules using data cubes. In Proc. 3rd Int. Conf. Knowledge Discovery and Data Mining #KDD'97#, pages 207#210, Newport Beach, California, August 1997.

[27] R. J. Bayardo. E#ciently mining long patterns from databases. In Proc. 1998 ACM-SIGMOD Int. Conf. Management
of Data, pages 85#93, Seattle, Washington, June 1998.

[28] J. Han, G. Dong, and Y. Yin. E#cient mining of partial periodic patterns in time series database. In Proc. 1999 Int. Conf. Data Engineering #ICDE'99#, pages 106#115, Sydney, Australia, April 1999.

[29] G. Dong and J. Li. E#cient mining of emerging patterns: Discovering trends and di#erences. In Proc. 5th Int. Conf. Knowledge Discovery and Data Mining #KDD'99#, pages 43#52, San Diego, August 1999.

[30] J. R. Quinlan "Induction of decision trees" Machine Learning 1, pp 81-106, 1986.

[31] J.R Quinlan (1979): "Discovering Rules by Induction from large Collection of Example" in D. Michie (ed.): Experts Systems in the Micro Electronic Age, Edinburgh University Press.