Auto-label Threshold Generation for Multiple Relational Classifications based on SOM Network

Ram Prakash Gangwar
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

Jitendra Agrawal
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

Varsha Sharma
School of IT, Rajiv Gandhi Technical University, Bhopal (M.P.), India

ABSTRACT
Classification and Association rule mining are two basic tasks of Data Mining. Classification rules mining finds rules that partition the data into disjoint sets. This paper is based on MrCAR (Multi-relational Classification Algorithm) and Kohonen's Self-Organizing Maps (SOM) approach. SOM is a class of typical artificial neural networks (ANN) with supervised learning which has been widely used in classification tasks. For small disjunction mining, we collocate with a new auto level threshold generation method in our algorithm to solve the problem of unclassified data of MrCAR. So, we optimize the classification rate of MrCAR with SOM network and improve the efficiency of classification. This approach is highly effective for classification of various kinds of databases and has better average classification accuracy in comparison with MrCAR. Finally the results convincingly demonstrated that our proposed algorithm has high accuracy.

Keywords
Classification; Data mining; MrCAR; and SOM.

1. INTRODUCTION
Classification is an important subject in data mining and machine learning, which has been studied extensively and has a wide range of applications. Classification based on multiple relational association rules, also called multiple associative classifications is a technique for building accurate and efficient classifier. Classification is to build a model (called classifier) to predict future data objects for which the class label is unknown. Associative classification was proposed by Bing Liu et al [1], which called CBA Algorithm. It refers to Srikant and Agarwal proposed apriori association rule [2][3].The apriori association rule is a way to deal with transaction data of associative mining. Associative classification generally contains two steps: first it finds all the class association rules (CARs) whose right hand sides is a class label, and then selects strong rules from the CARs to build a classifier. In this way, associative classification can generate rules with higher confidence and better understandability compared to traditional approaches. Thus associative classification has been studied widely in both academic world and industrial world, and several effective algorithm [1][4][5][7] have been proposed successively. Subsequently, Wenmin Li et al. proposed CMAR algorithm [6]. The main difference is adding a concept of multiple class-association rules in CMAR algorithm. In short, it is using two or more rules for predicting a class of test instance, and it replaces traditional method of using only one rule in classification phase. After calculating a weight, classifier will take the highest weight of class group, and then using it classifies a test instance. Finally, it had been demonstrated that using multiple rules classification can obtain good accuracy. Since 1980 Robert C.Holte argued the problem of small disjunction. The small disjunction problem means that a rule may be important for some instances but own low support so that the rule would be removed in rule mining phase. So according to this problem the accuracy may be decreased [8].CMAR used support and confidence been association rule threshold and it lies in support characteristic of downwards-closure to reduce calculation cost, so as to cut down the training cost. But it may cause a problem that some important rules may be losing, so that some instances may not be able to classified correctly. If we decrease minsup to find out these low support rules, then we meet a large no. of rules and noise rules to be generated. However the existing multiple association classification algorithm do not good accuracy which are used classify accurately. In the successive manner, another approach *MrCAR*, for associative classification which can be applied in multi-relational data environment. The main idea of MrCAR is to mine relevant *features* of each class label in each table respectively, and generate strong classification rules. By relevant features, we mean two kinds of frequent close itemsets: single table itemsets in the target table and cross table itemsets in non-target tables. In this paper we propose new classification techniques through integration of MrCAR and Self Organizing Map (SOM) network to improve the classification accuracy. This algorithm consist of two steps: firstly find out the association rules with the help of frequent pattern mining and correlations among these rules to classify the data set and then auto-label threshold generation in the form of winner of the trainer and learner itemsets of the Self Organizing Map (SOM) network. Kohonen's Self-Organizing Map (SOM) [9] tries to emulate the development of topological maps in the brain using locally interconnected networks and an algorithm based on local neighborhoods. The cerebral cortex of the brain is arranged as a two-dimensional plane of neurons and spatial mappings are used to model complex inputs. These maps can successfully approximate high-dimensional input spaces by extracting invariant features of the input signals and maintaining topological relationships between them in lower dimensions. This means that topological relationships in external stimuli are preserved and complex multidimensional data can be represented in a lower dimensional space. In other words, the SOM can produce an ordered low dimensional representation of an input data space. Typically such input data is complex and high-dimensional with data elements being related to each other in a nonlinear fashion.

.

2. METHODS

2.1 CMAR

CMAR algorithm was proposed by wenmin li et al 2001[6][10]. The major feature is using multiple rules to classify an instance. In the past, the first thing is to sort all of the rules of classification in associative classification. The sorting priority is ordered by confidence>support>length of rule. It is an order to evaluate what rules have own the highest priority for using. In classification phase, the first thing is to scan rules from high priority to low, and then take the first matched rule to classify test instance. After finding the first matched rule, we will use the rule's class to classify test instances. However CMAR algorithm is unlike single rule classification method. It adopts multiple class association rules to classify test instance.

This algorithm consists of two phases: Rule generation and Classification.

In first phase rule generation, CMAR computes the complete set of rules in the form of

$$R: P \rightarrow C$$

Where P is a pattern in the training data set and C is a class label such that sup(R) and conf(R) pass the given support and confidence thresholds respectively follows ,CMAR prunes some rules and only select a subset of high quality rules for classification.

In the second phase classification, CMAR is a multiple class association rules algorithm. When a test instance will be classified, CMAR algorithm will select a member of rules. Then, pick out the rule group which one of the rule groups has own the highest weighted chi-square ($X^{2)}$ to classify test instances. Trivially, if all the rules matched the new object have the same class label, CMAR just simply assigns that label to the new object if the rules are not consistent in class labels, MAR's divides into groups according to class labels. All rules in a group share the same class label and each group has a distinct label. To compare the strongest group, CMAR algorithm will calculate their weighted X^2. Then the class of the maximal weighted X^2 of group is selected to classify test instances.

$$\text{weighted } X^2 = \sum \frac{X^2 X^2}{\max X^2}$$

$$\max X^2 = \left(\min\left\{ \sup(p), \sup(c) \right\} - \frac{\sup(p)\sup(c)}{|T|} \right)^2 |T| e$$

where

$$e = \frac{1}{\sup(p)\sup(c)} + \frac{1}{\sup(p)(|T| - \sup(c))} + \frac{1}{(|T| - \sup(p))\sup(c)} + \frac{1}{(|T| - \sup(p))(|T| - \sup(c))}$$

Max X^2 computes the upper bound of X^2 value of the rules. In the areas of weighted calculation, CMAR algorithm adds X^2-testing conception called weighted X^2

2.2 SOM (kohonen's self-organizing map)

A self organizing map (SOM) or self organizing feature map[9] is a type of artificial neural network that is trained using unsupervised learning to produce a low dimensional (typically two dimensional),disritized representation of the input space of the trained samples, called a map. Self organizing maps are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space.

On the role of self organizing map among neural network model

The network architecture and signal processes used to model nervous system can roughly be divided into three categories, each based on a different philosophy. *Feed forward networks* [11] transform sets of input signals into sets of output signals. The desired input-output transformation is usually determined by external, supervised adjustment of the system parameters. In *feedback networks* [12], the input information defines the initial activity state of a feedback system, and after state transitions the asymptotic final state is identified as the outcome of the computation. In the third category, neighboring cells in a neural network compete in their activities by means of mutual laterals interactions, and develop adaptively into specific detectors of different signal patterns. In this category learning is called competitive, unsupervised or self organizing

Utility

- Visualizing N dimensional data in 2D
- Detecting similarity and degree's of similarity

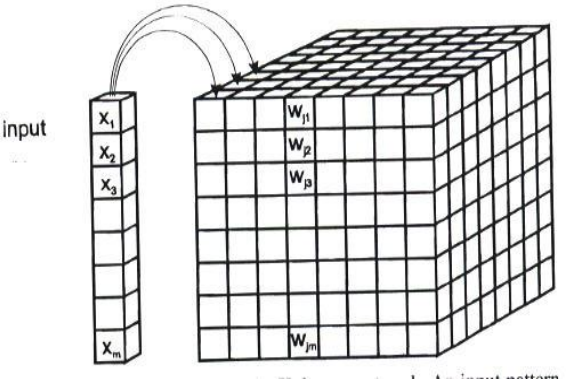

Fig.1: Structure of the map

The given figure (1) [13] shows a square grid, each grid point is a vector containing the descriptor values. The grid wraps round the edges and the grid is initialized with random vectors

Training the Map

- Each descriptor vector in the training set is presented to all grid points.
- Select the closest matching grid point based on minimum Euclidean distance

$$d(j) = \sqrt{\sum_{i=1}^{m} (x_i - w_{ij})^2}$$

- Modify the selected grid point and it neighbors.
- Degree of modification reduces with each training iteration.

Specific neighborhood of j and for all i, calculate the new weight.

$$w_{ij}(new) = w_{ij}(old) + \alpha[x_i - w_{ij}(old)]$$

The Learning Factor
α (t) is termed the learning factor. Some properties of the function are:

1. $0 < \alpha < 1$
2. Decreases monotonically with time
3. When $\alpha = 0$, learning stops
4. α can be varied in several ways

- Constant decrement
- Function of training iteration
- Recursive -

$$\alpha(t+1) = 0.5\alpha(t)$$

2.3 MrCAR (multi-relational associative classification algorithm)

A multi-relational associative classification algorithm has the idea of *class recurrent closed itemset* [17]. A CRCI is a recurrent closed itemset which contains a class label and an itemset, so it at least contains 2 items. Only CRCIs do we care because they reproduce the associations between class labels and other itemsets. By means of these itemsets we can produce classification rules. The Algorithm has three steps-

Step 1: Drawing out multi-relational CRCIs.
Step 2: generate multi-relational classification rules.
Step 3: Predicting class labels bases on the rules.

First of all, we need to extract CRCIs from multi-relational datasets. The algorithm for drawing out CRCIs is described below

```
Function GenCRCIs
  Main steps:
1   for each table T in the input database
2       produce a set of initial nodes: ITPairs;
3       Charm Extend (ITPairs, C);
4       for each itemset c in C
5         CRCI = Union (c, label);
6         if (T is the target table)
7       CRCI.tids = Intersection (c.tids, label.tids);

8         sup Count = CRCI.tids.size ();
9   else                    //T is a non-target table
10        supCount = 0;
11        for each tuple t in c.tids
12          t has a set of target tuple IDs: IDSets;
13          get the number of labels in IDSets: n;
14      supCount += n;
15   if (supCount satisfy minsup) keep the CRCI;
```

This Algorithm shows the main steps to mine CRCIs. Line 1-2 represents the initialization of the algorithm. In line 3 algorithm CHARM [14][15] to discover all of the recurrent closed itemsets and store them in *C*. Thus we have generated all the recurrent closed itemsets without class labels. Line 4-5 combines each recurrent closed itemset with a class label to produce candidate CRCIs. Line 6-8 calculates the support count (denoted as supCount) of a particular table CRCI. A candidate CRCI is got by combining a recurrent closed itemset c in a non-target table with a class label in the target table. The key is to calculate the support count of the candidate CRCI to check if it satisfies minsup. Line 9-14 shows the process. Support count is initialized to zero. For an itemset c and a class label, we first get its tids which correspond to a set of non-target tuples containing c. Line 12-14 deals each such tid as follows: for each nontarget tuple we have a set of target tuple IDs through the method of tuple ID propagation. The number of the tuples in the set that contain the class label is counted and is added to supCount. We repeat the steps in Line 12-14 until all tuples in c's tids have been processed. Thus we can get the support count of a cross table CRCI and check if we need keep it or not. After the drawing out multi-relational CRCIs in a relational database, it's time to generate classification rules from CRCIs. A classification rule is an implication of the form X→L, where X (called *head*) is an itemset and L is a class Label. The *confidence* of a classification rule, conf (X→L), is (sup (XL)/sup (X))*100%. Our target is to produce all the classification rules that satisfy the user-specified minimum confidence threshold (called *minconf*) from CRCIs. We perform classification on a set of test tuples and evaluate the performance of the rules produced in the last step. In MrCAR, tuples are classified one by one. Generally one tuple could be covered by multiple classification rules, say n rules. The n rules are ranked by their confidences and build a classifier (R_1, R_2… R_n, Default Class), where conf (R_i) ≥ conf (R_j) (i<j). In classifying a test tuple, the rule with the highest confidence will classify it. If there are multiple rules with equal highest confidence, we utilize voting. If no rule satisfies the test tuple, the default class, which is the majority class in the training set, is selected as the class label we predict.

3. PROBLEM STATEMENT

Classification on a set of test tuples and estimate the performance of the rules produced. Due to the study of previously proposed multi-relational classification algorithms shows that in classification of multi-dimensional association rule mining, classification faced problem of continuity of frequency rules [18]. The multiple classification rules are approved by their support and confidence, which are used to build a classifier. In classifying the test tuples, the rules with the maximum confidence will classify it. But these entire algorithm Based on support and confidence threshold framework, which arising small disjunction mining problem. Due to Generation of weight and maintained of weight value of support difficult [19]. Therefore, we collocate with a new auto level threshold generation method in our algorithm to solve the problem of small disjununction mining. So, we optimize the classification rate of MrCAR with SOM network approach to classification of assorted kinds of databases. Finally the results demonstrated that our proposed algorithm has high accuracy. If there are multiple rules with equal maximum confidence then we use selection. If no rules satisfy the test tuples the default class, which is the majority class in the training set, is selected as class label.

4. EXPERIMENTAL RESULT

In this paper we have used UCI multivariate characteristics wine dataset. This wine dataset attribute characteristics are real and integer for associated task of classification. The experiments were performed on Windows 7, 64 bits and Intel i3 core (350) and 3GByte RAM. The program was developed by MATLAB 7.8(2009) version. The experiments point of view settings are according to the most common settings in associative classification algorithm. The support threshold (minsupport) is set to 1% and confidence (minconfidence) is varied from .1 to 1% and optimizes the accuracy and

efficiency on both methods, MrCAR and MrCAR-SOM.We compare MrCAR with MrCAR-SOM in our Experiments.

Where MrCAR-SOM is our implemented program which is execute in multi-relational environment and shows that MrCAR-SOM has high accuracy.

Algorithms	MrCAR	MrCAR-SOM
Accuracy	82.6748	90.6706
Efficiency	1.8298	2.4039
Generated Rules	40	40

Table. 1: Comparison between MrCAR, MrCAR-SOM

From the experiment we can see that in database MrCAR-SOM predicts more precisely than MrCAR. It validates again that multi-relational classification algorithm with SOM network has higher accuracies than traditional algorithms. However MrCAR is not as efficient as MrCAr-SOM. The results are shown in figures below.

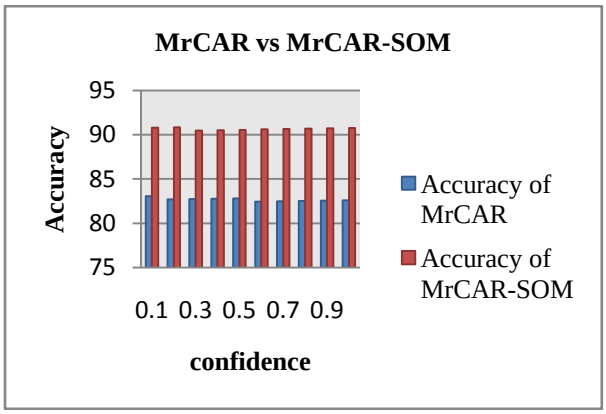

Fig. 2: Comparison of accuracy between MrCAR and MrCAR- SOM on different confidence.

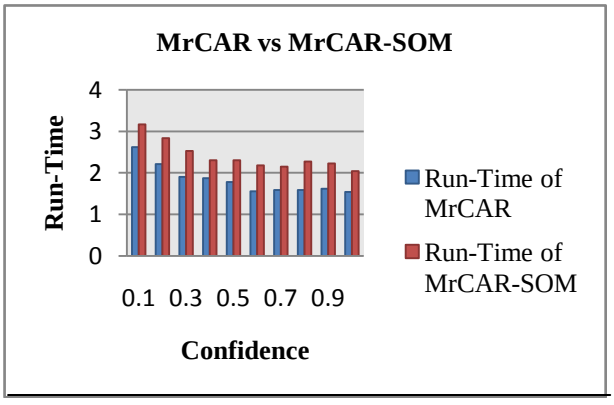

Fig. 3: Comparison of efficiency between MrCAR and MrCAR-SOM on different confidenc.

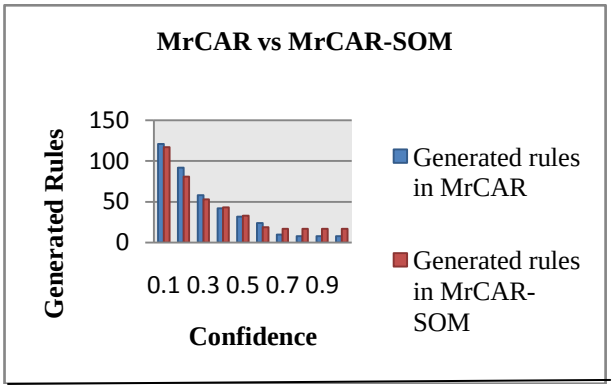

Fig. 4: Comparison of generated Rules between MrCAR and MrCAR-aSOM on different confidence.

From the above figures,we can see that MrCAR-SOM achives excellent accuracy. It is more efficient that MrCAR approach. Our proposed classification algorithm accuracy is 90.6706, which accuracy is 7.9958 greater than MrCAR algorithm accuracy 82.6748.

5. CONCLUSION AND FUTURE WORK

This paper, introduced a different associative classification algorithm in multi-relational environment, MrCAR-SOM. Experiment results show that MrCAR-SOM gets higher accuracy and efficiency compared to existing multi-relational algorithm. The rules produced by MrCAR-SOM have a more comprehensive depiction of databases. In this paper we find recurrent itemset and produce classification rules with the help of support-confidence structure. It may discover more significant features of each class label by using related measures extending current framework.

6. REFERENCES

[1] Bing Liu, Wynne Hsu, Yiming Ma,"Integratin Classification and Association Rule Mining" *Proceedings of the Fourth* International *Conference on Knowledge Discovery and Data Mining (KDD-98)*, New York, USA, 1998, pp 80-86.

[2] Rakesh Agrawal, Tomasz Imieli_ski, and Arun Swami, "Mining Association Rules Between Sets of Items in Large Databases", ACM *SIGMOD Conference*, New York, USA, 1993, pp 207-216.

[3] Rakesh Agrawal, and Ramakrishnan Srikant, "Fast algorithms for mining association rules in large databases", *Proceedings of the 20th International Conference on Very Large Data Bases(VLDB)*, Santiago, Chile, September 1994, pp 487-499.

[4] G. Dong, X. Zhang, L. Wong, and J. Li, "CAEP: Classification by aggregating emerging patterns", *Proceedings of the 2nd International Conference on Discovery Science*, Springer-Verlag, Berlin Heidelberg, 1999, pp. 30-42.

[5] K. Wang, S. Zhou, and Y. He, "Growing decision trees on support-less association rules", *Proceedings of the KDD*, ACM, Boston Massachusetts, 2000, pp. 265-269.

[6] W. Li, J. Han, and J. Pei, "CMAR: Accurate and efficient Classification Based on Multiple Class-Association Rules", *Proceedings of the ICDM*, IEEE Computer Society, San Jose California, 2001, pp. 369-376.

[7] X. Yin, and J. Han, "CPAR: Classification based on Predictive Association Rules", *Proceedings of the SDM*, SIAM, Francisco California, 2003.

[8] Robert C. Holte and Liane E. Acker and Bruce W. Porter., "Concept Learning and the Problem of Small Disjuncts" *In Proceedings of the Eleventh International Joint Conference on Artificial Intelligence*, 1989, pp 813-818.

[9] t. Kohonen. Self-organizing maps. Springer, Berlin, Heidelberg, 2001. Third extended edition, ISBN 3-540-67921-9.

[10] w. Li, "classification based on multiple association rules," *m.sc. Thesis*, Simon Fraser University, April 2001.

[11] D.E .Rumelhart, G.E .Hinton, and R.J. Williams ″learning Internal representation by error prorogation,″inparallelDistributedprocessing.vol.1.:Foundations,D.E.Rumelhart,J.L.McClelland and the PDP research group,Eds.Cambridge,Mass.:MIT Press,1986,pp.318-362

[12] J.J.Hopfield, ″Neural network and physical systems with emergent collective computational activities,″Proc.Natl.Acad.Sci.USA,vol.79,pp.2554-2558,1982.

[13] Kohonen, T. *Self Organizing Maps;* Springer Series in Information Sciences Springer: Espoo, Finland, 1994.

[14] M.J. Zaki, and C.J. Hsiao, "CHARM: An Efficient Algorithm for Closed Itemset Mining", *Proceedings of SIAMOD International Conference on Data Mining*, 2002, pp. 457-473.

[15] M.J. Zaki, and K. Gouda, "Fast Vertical Mining Using Diffsets", *Proceedings of the 9th ACM SIGKDD*, ACM, New York USA, 2003, pp. 326-335.

[16] Jiawei Han, and Micheline Kamber, *Data Mining: Concepts and Techniques, Second Edition*, China Machine Press, Beijing, 2007. 260

[17] Yingqin Gu, Hongyan Liu, Jun He, Bo Hu and Xiaoyong Du,"MrCAR: A Multi-relational Classification Algorithm based on Association Rules", 2009 International Conference on Web Information Systems and Mining.

[18] Saˇso Dˇzeroski," MultiRelational Data Mining: An Introduction" Jamova 39, SI1000 Ljubljana, Slovenia

[19] pei-yi hao, yu-de chen," a novel associative classification algorithm: a Combination of lac and cmar with new measure of Weighted effect of each rule group", Proceedings of the 2011 International Conference on Machine Learning and Cybernetics, Guilin, 10-13 July, 2011