Journal of Theoretical and Applied Information Technology
10th July 2014. Vol. 65 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

ENHANCED TECHNIQUES FOR ASSOCIATION RULE CLASSIFICATION IN LARGE DATASETS

[1]MINI.T.V, [2]Dr.R.NEDUNCHEZHIAN

[1]Asst.Professor, Dept. Of Computer Science, Sacred Heart College, Chalakudy

[2] Professor, Dept. Of CSE, Sri Ranganathar Institute of Engineering and Technology, Coimbatore

E-mail: [1]sistermiranto@gmail.com, [2]rajuchezhian@gmail.com

ABSTRACT

Associative classification performs knowledge discovery by combining the tasks of the Knowledge Discovery in Databases (KDD), association rule mining and classification. It is a promising method that improves classification in terms of accuracy when compared with traditional classification methods. The major issues with associative classification are its inability to work efficiently with huge datasets and the huge number of association rules generated. In this paper, these issues are addressed through the use of hierarchical partitioning with Frequent Pattern List and multiple projections of pruning algorithms with optimized rule selection procedure. Experimental results prove that these solutions provide the advantage of able to work with both small and large sized datasets without degrading the classification accuracy. The proposed algorithm shows an average efficiency gain of 47.90% with respect to reduction in number of rules, while showing a notable improvement in the classification accuracy.

Keywords : *Association Rule, Classification, Large Datasets, Frequent Pattern List, Hierarchical Partitioning.*

1. INTRODUCTION

Associative Rule Classification(ARC) is a popular and well-researched method that has attracted several academicians and research scholars [1] and the interest in this field is still active, owing to the fast development of innovative methods in the field of information technologies, storage facilities, processing units along with the users' ability to generating and collecting data. ARC combines two important data mining tasks, namely, associative rule discovery and classification, to build a classification model that can perform prediction.

The first step of an ARC discovers the complete set of Class Association Rules from the training dataset, from which a subset (optimal rules) are selected to build the classifier. Selection of optimal rules is performed using rule pruning algorithms like Database Coverage or lazy pruning algorithm [2]. After the construction of ARC, its predictive power can be evaluated using the test data objects to predict their class labels. This paper focuses on techniques that improve the process of associative rule discovery and classification steps of ARC. Two main issues of associative rule discovery is its degraded performance with large databases and number of association rules created. Both these issues degrade the performance of prediction (or classification) and several techniques have been addressed to solve these issues ([3];[11];[12]). However, even these data structures reports 'out of memory' while handling large datasets [9]. Thus, the scalability issue is a major problem with associative rule discovery.

To solve this problem, a Hierarchical Partitioning (HP) method was proposed by [10], where a data structure called Frequent Pattern List (FPL) was used to improve the speed and make it adaptable to huge transaction databases. This structure was used with FP-Growth algorithm to discover association rules. While this method (referred to as HP-FPL method in this paper) handled the problem of scalability, the performance depended on the correct selection of the threshold minimum support (min-supp). In this paper, the HP-FPL is extended to use an automated process that estimates the min-supp value without user intervention. Further, the parameter-less HP-FPL algorithm for ARC was extended to use multiple projection pruning algorithms to improve classification performance. The enhanced ARC algorithm is termed as HM2ARC (Hierarchical partitioning with FPL with Multiple projection pruning and 2-step Associative Rule Classification). The rest of the paper is

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

organized as follows. Section 2 gives details regarding the proposed HM2ARC algorithm. Section 3 presents the experimental results while Section 4 concludes the work with future research directions.

2. METHODOLOGY

The steps involved in the proposed HM2ARC (Hierarchical partitioning with FPL, Multiple projection pruning and 2-step Associative Rule Classification) is presented in Figure 1.

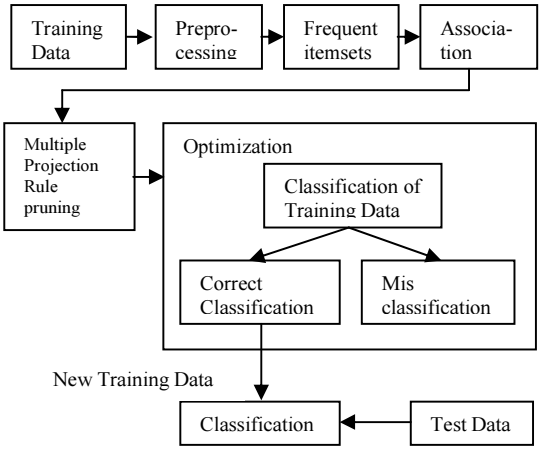

Figure 1 : Steps in HM2ARC

2.1. Preprocessing

The algorithm begins by using an automated process to estimate minimum support, using which the Hierarchical Partitioning with Frequent Pattern List finds the frequent itemsets and generates the association rules. This step is considered as a preprocessing step in this work and is explained in this section. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a set of items. Let 'D' be a set of transactions, where each transaction 'T' is a set of items such that $T \subseteq I$. The main aim of association rule mining is to discover rules that have support and confidence greater than a user-specified minsup(minimum support) and minconf(minimum confidence). An association rule is defined as a relationship between itemsets and the general form is given as $A \rightarrow B$, where A and B are said to be frequent itemsets and $A \subset I$, $B \subset I$ and $A \cap B = \emptyset$. In the equation, the itemsets, A and (B – A) are called the antecedent and consequence of the rule 'r' respectively. The rule $X \rightarrow Y$ holds in the transaction set T with confidence 'c', if c% of transactions in T that support 'X' also support 'Y'. The rule has support 's' in T if s% of the transactions in T contains $X \cup Y$. In general, the support and confidence values are estimated using the formulae in Equation (1).

$$Support(X) = \frac{|\{t \in D \mid X \subseteq t\}|}{|D|}$$
$$Confidence(r) = \frac{Support(Y - X)}{Support(X)} \quad (1)$$

The performance of frequent mining and association rule generation depends on two parameters, minimum confidence (min-conf) and minimum support (min-supp), which are user-specified. Correct selection of these two parameters is critical and is highly dependent on the nature of database. In general, a high min-supp will result with very few association rules, while a low min-supp will generate very high association rules and both these situations degrade the performance of associative mining. This paper provides an automated method to solve the problem of user-supplied minimum support and minimum confidence thresholds using polynomial function.

Let D be the database with n itemsets (i_1, i_2, …) and let S be the support of each item which is distributed in an interval [a, b], where a and b are the minimum and maximum support in D. Let M be the maximum number of items in D. The proposed method, termed as AMSM (Automated Minimum Support Method), first scans D to estimate S of each item, from which the average support is calculated (Equation 2). This average value is used as initial min-supp which is then used during the optimal minimum support value estimation.

$$A_s = \sum_{i=1}^{n} S \Big/ n \quad (2)$$

The aim of AMSM is to determine min-supp of D within the interval $[M_i, M_a]$, which can be implemented as a mapping $f : [M_i, M_a] \rightarrow [0, 1]$. Here, M_i is the minimum and M_a is the maximum support in D. As, in many cases, this mapping is hidden, it is necessary to find an approximate polynomial approximation function $f^{\#}$ for f as given in Equation (3). Here, let X in $[M_i, M_a]$ be $x_1, x_2, \ldots, x_n$ and Y in [0, 1] be $[y_1, y_2, \ldots, y_n]$ as listed below. The proof of the equations are given by [13] and [14].

$$f(x) = \frac{1}{b^n - a^n} x^n + \frac{a^n}{a^n - b^n} \quad (3)$$

Journal of Theoretical and Applied Information Technology
10th July 2014. Vol. 65 No.1

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

X	x_1	x_2	…	x_n
Y	y_1	y_2	…	y_n

Consider a transaction dataset, D, as given in Table I with its corresponding itemsets and support as given in Table II. From these tables, it can be understood that $M_i = 0.25$ and $M_a = 0.75$ and A_s is 0.5. Substituting these values in linear function (Equation (2), $f(A_s)$ is obtained as in Equation (4). From this value, the actual optimal min-supp of D is estimated as in Equation (5). From the support value calculated the minimum confidence measure is estimated using the lift measure. Using this automatically estimated min-supp value on D, it can be found that the frequent itemsets are A, B, C, D, AB, AC, BC, CD.

$$f(As) = \frac{1}{0.75 - 0.25}x + \frac{0.25}{0.25 - 0.75} = 2x - 0.5 \quad (4)$$

$$min\text{-}supp = f^{-1}(A_s) = \frac{f(A_s)+0.5}{2} = \frac{0.5+0.5}{2} = 0.5 \quad (5)$$

Table I : Transaction Database (D)

A	B	C	
	B	C	D
A		C	D
A	B		

Table II : Itemsets and Support(D)

A	0.75	AD	0.25
B	0.75	BC	0.50
C	0.75	BD	0.25
D	0.50	CD	0.50
AB	0.50	ABC	0.25
AC	0.50		

2.2. Frequent Itemsets and Rule Generation

The HP-FPL algorithm was designed to address the issue of scalability with frequent pattern mining algorithms. The algorithm made use of a data structure called Frequent Pattern List (FPL), which is a linear list of item nodes, where each node represents one frequent item in the transactional database. To handle the problem of large datasets, a hierarchical partitioning algorithm was used. The algorithm uses a modified version of FP-Growth algorithm and detailed description of the hierarchical partitioning algorithm along with its mining algorithm can be found in [10].

2.3. Rule Pruning

This section presents Multiple Projection Pruning Algorithm (M2PA), which is used mainly to reduce the high number of association rules generated by HP-FPL without any degradation in performance of HP-FPL. Two frequently used pruning algorithms, namely, Database Coverage Pruning (DCP) and Chi-Square Pruning (CSP) are used. A simple Support-Confidence based Algorithm (SCP) is also used. The main aim of M2PA is to apply SCP and CSP (or DCP) in succession to improve mining task. The flow of the M2PA is shown in Figure 2.

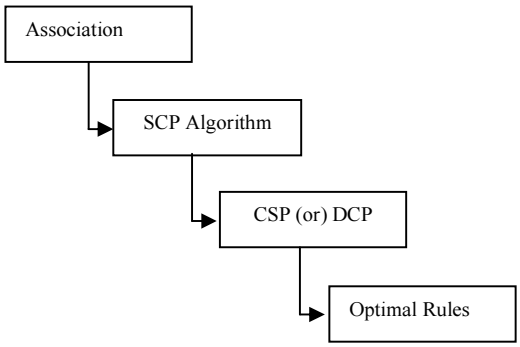

Figure 2 : Steps in M2PA

2.3.1. SCP Algorithm

This algorithm starts by arranging the rules in ascending order of support and confidence thresholds. This method used normally as a ranking algorithm during associative classification process, is modified as a pruning algorithm in this paper. This algorithm is based on the assumption that all rules having high support and confidence values will have quality information, while the rest of the rules are valueless and hence can be discarded. Quality rules are identified using the procedure given in Figure 3, where C, S, N represents the confidence, support and number of attributes in the left hand side of the rule and R1 and R2 are two association rules. Using this method, quality rules are assigned higher ranks. After ranking process, the rules are sorted in descending order and the pruning process begins. During pruning process, each rule should satisfy the property of generality. After ranking, the top-K rules are taken as optimal rules. The K value is estimated by using the rule of thumb [7] method (Equation 6), where n is the number of association rules. The result of SCP algorithm is a set of rules arranged in descending order according to its rank and is used as input by the CSP or DCP algorithm.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

$$k = \sqrt{\frac{n}{2}} \quad (6)$$

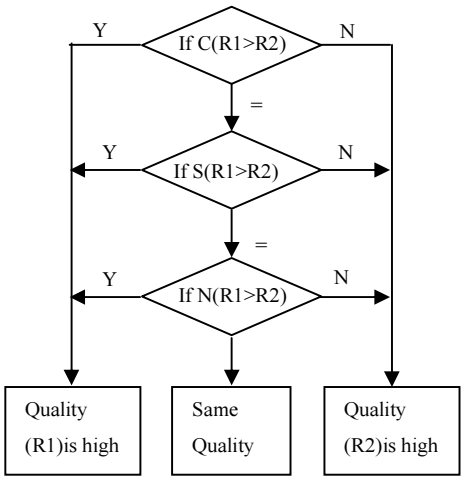

Figure 3 : SCP Algorithm

2.3.2. DCP Algorithm

Database Coverage Pruning technique is a well known technique that was proposed by [5] and later adapted by [6] to improve the process of association rule classification. The procedure of database coverage is presented in Figure 4, where R is the set of association rules generated by SCP algorithm and T is the training dataset. This method tests the generated rules against the training data set, and only high quality rules that cover at least one training instance not considered by other higher ranked rules are kept for later classification.

2.3.3. CSP Pruning

Chi-square testing (χ^2) is a well-known discrete data hypothesis testing method from statistics, which evaluates the correlation between two variables and determines whether they are independent or correlated [8]. The test for independence, when applied to a population of data, determines whether they are positively correlated or not (Equation 7). where e_i is the expected frequencies and f_i is the observed frequencies.

$$\chi^2 = \sum_{i=1}^{k} \frac{(f_i - e_i)^2}{e_i} \quad (7)$$

When the expected frequencies and the observed frequencies are different, the hypothesis that they are correlated is rejected. This method has been used in associative classification to prune negatively correlated rules. For example, a test can be done on every discovered rule, such as $r : x \rightarrow c$, to find out whether the condition x is positively correlated with the class c. If the result of the test is larger than a particular constant, there is a strong indication that x and c of r are positively correlated and therefore r will be stored as a candidate rule in the classifier. If the test result indicates negative correlation, r will not take any part in the later prediction and is discarded.

```
For each rule r_i in R Do
      Find all applicable data instances in T that match r_i's
      condition
   If r_i correctly classifies a training instance in T
      Mark r_i as a candidate rule in the classifier
      Remove all instances in T covered by r_i
   end if
   If r_i cannot correctly cover any training instance in T
      Remove r_i from R
   end if
End for
```

Figure 4 : Database Coverage

2.4. Optimization after Pruning

While the M2PA reduces the number of rules generated, the number of rules produced is still on the higher side while used with larger datasets. The challenge now is to eliminate more rules without compromising the classification accuracy. In this study, the performance of each rule in the training set is evaluated and pruning is again performed based on the rule performance on the training set. The True Positive (TP), True Negative (TN), False Positive (FP) and False Negative (FN) of the training set are calculated. Here FP and FN are the number representing misclassified rules, while TP and TN represented correctly classified rules. The proposed optimization process creates a new training set by only considering the correctly classified rules, thus increasing the quality of rules selected for classification.

2.5. Classification

The next step of ARC is the actual classification, which can be performed using the procedure presented in Figure 5. During the training phase, the set of association rules generated using the techniques mentioned in the previous sections, are used. This set of rules is denoted as R. During testing (denoted as O), the classification process searches the rule set for finding a class that is close to the new rule. The prediction puts O into a class that has the highest confidence sum. That is, the labeling of O is performed by attaching the test data to a class that most matches the rules generated.

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

```
S = NULL;
for each rule r in R
        if (r ⊂ O) {count++}
            if (count == 1)
                fr.conf = r.conf
                 /* keep the first rule confidence*/
                S = S ∪ r
            else if (r.conf > fr.conf)
                S = S ∪ r
            else exit
        end if
  divide S in subsets by category S1, S2, … , Sn
  for each subset S1, S2, … , Sn
        sum the confidences of rules and
        divide by the number of rules in Sk
  Predict class for O
End for
```

Figure 5 : Classification Process

3. EXPERIMENTAL RESULTS

The performance evaluation of the algorithms was conducted using 10 datasets obtained from UCI repository [4]. The selected datasets are listed in Table III, where A_n denotes the number of attributes, I_n is the number of instances and C_n represents the number of classes. The first set of experiments was conducted to evaluate the pruning and optimization algorithms used to produce quality classification rules from training data. A 10-fold cross validation method was used. The number of association rules before and after applying the pruning and optimization algorithms is presented in Table IV.

From the results it can be seen that the multiple projection pruning algorithm combining SCP with DCP algorithm produces better results than CSP algorithm. Hence the optimization procedure was combined with M2PA using SCP and DCP algorithm. It is further clear from the results that the M2PA when combined with optimization produces small number of classification rules with all datasets. It is also interesting to note that the percentage of reduction is high with large sized datasets than small sized datasets.

The reduction gain obtained by M2PA combined with optimization procedure with respect to number of association rules generated when compared to DCP algorithm. The reduction gain obtained by Iris is 14.29%, Diabetes is 12.20%, Pima is 5.47%, Breast is 7.53%, Segment is 75.14%, German is 65.60%, Waveform is 81.01%, Sick is 67.64%, Chess is 60.44% and Splice is 89.68%. From this percentage result, it can be seen that the percentage gain obtained with large datasets is greater than small datasets, thus proving the success of using the algorithm with large sized datasets.

Table III : Datasets Used

Dataset	A_n	I_n	C_n	Dataset	A_n	I_n	C_n
Iris	4	150	3	German	20	1000	2
Diabetes	8	768	2	Waveform	21	5000	3
Pima	8	768	2	Sick	25	3163	2
Breast	9	699	2	Chess	36	3196	2
Segment	19	2310	7	Splice	60	3190	3

Table IV : Number of Association Rules Generated

Dataset	Without Pruning	DCP	M2PA +DCP	M2PA +CSP	M2PAwith Optimi-zation
Iris	135	35	30	33	27
Diabetes	4086	205	180	191	120
Pima	4083	201	190	196	119
Breast	16738	146	135	139	116
Segment	106558	1798	1397	1418	447
German	128976	1977	1278	1291	680
Waveform	130089	3680	1890	1899	759
Sick	141243	2157	1573	1586	712
Chess	175365	2017	1855	1861	731
Splice	789745	9987	1283	1292	568

As the main aim of associative classifiers is to use small set of quality rules to achieve maximum classification accuracy, the performance analysis compares the proposed HM2ARC with CBA, CMAR, C4.5 and SVM classifiers. The results obtained are presented in Table V.

Table V : Accuracy (%)

Dataset	CBA	CMAR	HM2ARC
Iris	94.70	95.00	95.72
Diabetes	74.50	75.80	75.96
Pima	74.90	75.10	76.66
Breast	96.30	96.40	96.91
Segment	88.46	89.12	90.45
German	90.64	91.02	91.98
Waveform	91.44	91.89	92.13
Sick	89.78	90.16	91.87
Chess	86.88	87.21	87.77
Splice	91.15	91.98	92.56

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

From the results pertaining to accuracy, it is clear that the proposed HM2ARC algorithm does not compromise on accuracy and maintains classification performance. On average, the proposed algorithm showed 1.49% and 0.93% efficiency gain over CBA and CMAR classifiers.

4. CONCLUSION

This paper presented a classification system based on association rules. First, during preprocessing, an automated algorithm that calculates the min-supp threshold for frequent itemset identification and association rule generation was proposed. In the second step, in order to handle large datasets, a hierarchical partitioning algorithm with frequent pattern list combined with fp-growth algorithm was used. To reduce the number of association rules generated by the second step, a multiple project pruning algorithm combining Top-K support-confidence algorithm and Database Coverage Pruning or Chi-Square Pruning technique was used. In the top-K pruning algorithm, the K value for identifying potential rules was estimated automatically. To further select quality rules without reducing classification accuracy, an optimized procedure was proposed to select only the correctly classified training rules. Finally, in the last step, classification was performed by finding a class that is close to the new rule. Experimental results proved that the proposed enhancement operations are successful, which was deduced by the high classification accuracy obtained by the proposed associative classifier. Strengthening the preprocessing steps to include noisy data removal and missing values are to be included in future. Further, techniques to improve the classification process also have to be probed.

REFERENCES

[1]. Aggarwal, C.C., Li, Y., Wang, J. and Wang, J. (2009) Frequent pattern mining with uncertain data, International Conference on Knowledge Discovery and Data Mining, Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, Pp. 29-38.

[2]. Baralis, E. and Garza, P. (2002) A Lazy Approach to Pruning Classification Rules, Proceedings of the 2002 IEEE International Conference on Data Mining (ICDM02) EEE Computer Society, P. 35.

[3]. Han J, Pei J, Yin Y (2000) Mining frequent patterns without candidate generation, Proceeding of the 2000 ACM-SIGMOD International Conference on Management of Data (SIGMOD00) Dallas, TX, Pp. 1-12.

[4]. http://archive.ics.uci.edu/ml/datasets.html, Last Accessed during April, 2014.

[5]. Li, W., Han, J. and Pei J. (2001) CMAR: Accurate and efficient classification based on multiple class-association rules, IEEE International Conference on Data Mining (ICDM'01), San Jose, CA, Pp. 369–376.

[6]. Liu, B., Hsu, W. and Ma, Y. (1999) Mining association rules with multiple minimum supports, Proceedings of fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, California, Pp. 337-341.

[7]. Mardia, K.V., Bibby, J.M. and Kent, J.T. (1979) Multivariate analysis, Academic Press, New York.

[8]. Snedecor, W. and Cochran, W. (1989) Statistical Methods, Eighth Edition, Iowa State University Press.

[9]. Tseng, F.C. (2012) An adaptive approach to mining frequent itemsets efficiently, Expert Systems with Applications, Vol. 39, No. 18, Pp. 13166–13172.

[10]. Tseng, F.C. (2013) Mining frequent itemsets in large databases: The hierarchical partitioning approach, Expert Systems with Applications, Vol. 40, Issue 5, Pp. 1654-1661.

[11]. Tseng, F.C., Hsu, C. C. and Fu, K.S. (2005a) The frequent pattern list: Another framework for mining frequent itemsets, International Journal of Electronic Business Management, Vol. 3, No. 2, Pp. 104–115.

[12]. Tseng, F.C., Hsu, C.C. and Fu, K.S. (2005b) Efficiently mining frequent closed itemsets by eliminating data redundancies, Electronic Commerce Studies, Vol. 3, No.1, Pp. 39–56.

[13]. Zhang, C. and Zhang, S. (2002) Association rules mining: models and algorithms. Publishers in Lecture Notes on Computer Science, vol 2307, Springer Berlin, p. 243

[14]. Zhang, S., Wu, X., Zhang, C. and Lu, J. (2008) Computing the minimum-support for mining frequent patterns, Knowl Inf System, Vol. 15, Pp. 233-257.