Classification Based on Positive and Negative Association Rules

B.Ramasubbareddy rsreddyphd@gmail.com
Associate Professor,Dept.of CSE,
Jyothishmathi Institute of Technology& Science,
Karimnagar505001, India

A.Govardhan govardhan_cse@yahoo.co.in
Professor of CSE,
JNTUH College of Engineering,
Nachupally,Karimnagar,505001,India

A.Ramamohanreddy ramamohansvu@yahoo.com
Professor of CSE, S.V.University,
Tirupati 517502, India.

Abstract

Association analysis, classification and clustering are three different techniques in data mining. Associative classification is a classification of a new tuple using association rules. It is a combination of association rule mining and classification. In this, we can search for strong associations between frequent patterns and class labels. The main aim of this paper is to improve accuracy of a classifier. The accuracy can be achieved by producing all types of negative class association rules.

Keywords: data Mining, Association Analysis, Classification, Positive and Negative Association Rules.

1. INTRODUCTION

Data mining algorithms aim at discovering knowledge from massive data sets. Association analysis, classification and clustering are three different data mining techniques. The aim of any classification algorithm is to build a classification model given some examples of the classes we are trying to model. The model we obtain can then be used to classify new examples or simply to achieve a better understanding of the available data. Classification generally involves two phases, training and test. In the training phase the rule set is generated from the training data where each rule associates a pattern to a class. In the test phase the generated rule set is used to decide the class that a test data record belongs to. Different approaches have been proposed to build accurate classifiers, for example, naive Bayes classification, Decision trees, and SVMs. Data mining community proposed Association Rule Mining based Classification. This approach is called Associative Classification produces transparent classifier consisting of rules that are straight forward and simple to understand. Associative classification based on association rule mining searches globally for all rules that satisfy minimum support and confidence thresholds. In associative classification the classifier model is composed of a particular set of association rules, in which consequent of each rule is restricted to classification class attribute. Many improvements have been done in associative classification approach in recent studies and experiments thereof show that this approach achieves higher accuracy than traditional approaches.

The traditional associative classification algorithms basically have 3 phases: Rule Generation, Building Classifier and Classification as shown in *Fig.1.* Rule Generation employ the association rule mining technique to search for the frequent patterns containing classification rules. Building Classifier phase tries to remove the redundant rules, organize the useful ones in a reasonable order to form the classifier and the unlabeled data will be classified in the third step. Some experiments done over associative classification algorithms such as CBA [26], CMAR [23] and

MCAR [28] state that the associative classification methods share the features of being more accurate and providing more classification rules.

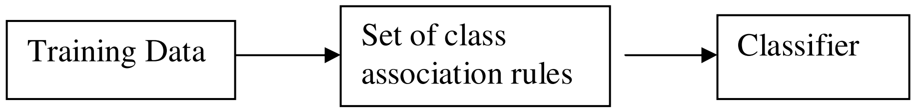

FIGURE 1: Associative Classifier

This paper is structured as follows: section II recalls preliminaries about Association Rules, In Section III, existing methods for associative classification are reviewed. The proposed algorithm is presented in Section IV and V. Section VI contains conclusions and future work.

2. BASIC CONCEPTS AND TERMINOLOGY

This section introduces association rules terminology and some related work on negative association rules and associative classification systems.

2.1 Association Rules

Let I = {i1, i2 ...in} be a set of items. Let D be a set of transactions, where each transaction T is a set of items such that $T \subseteq I$. Each transaction is associated with a unique identifier TID. A transaction T is said to contain X, a set of items in I, if $X \subseteq T$. An association rule is an implication of the form "$X \Rightarrow Y$", where $X \subseteq I$, $Y \subseteq I$, and $X \cap Y = \emptyset$. The rule $X \Rightarrow Y$ has a support s in the transaction set D if s% of the transactions in D contains $X \cup Y$. In other words, the support of the rule is the probability that X and Y hold together among all the possible presented cases. It is said that the rule $X \Rightarrow Y$ holds in the transaction set D with confidence c if c% of transactions in D that contain X also contain Y . In other words, the confidence of the rule is the conditional probability that the consequent Y is true under the condition of the antecedent X. The problem of discovering all association rules from a set of transactions D consists of generating the rules that have a support and confidence greater than given thresholds. These rules are called strong rules.

2.2 Negative Association Rules

A *negative association rule* is an implication of the form $X \rightarrow \neg Y$ (or $\neg X \rightarrow Y$ or $\neg X \rightarrow \neg Y$), where $X \subseteq I$, $Y \subseteq I$ and $X \cap Y = \Phi$ (Note that although rule in the form of $\neg X \rightarrow \neg Y$ contains negative elements, it is equivalent to a positive association rule in the form of $Y \rightarrow X$. Therefore it is not considered as a negative association rule. In contrast to positive rules, a negative rule encapsulates relationship between the occurrences of one set of items with the absence of the other set of items. The rule $X \rightarrow \neg Y$ *has* support s*% in* the data set s, if *s %* of transactions in *T* contain itemset *X* while do not contain itemset *Y*. The support of a negative association rule, $supp(X \rightarrow \neg Y)$, is the frequency of occurrence of transactions with item set *X* in the absence of item set *Y*. Let *U* be the set of transactions that contain all items in *X*. The rule $X \rightarrow \neg Y$ holds in the given data set (database) with confidence *c*, if c% of transactions in *U* do not contain item set *Y*. Confidence of negative association rule, $conf\ (X \rightarrow \neg Y)$, can be calculated with $P(X \neg Y)/P(X)$, where $P(.)$ is the probability function. The support and confidence of itemsets are calculated during iterations. However, it is difficult to count the support and confidence of non-existing items in transactions. To avoid counting them directly, we can compute the measures through those of positive rules.

3. RELATED WORK IN ASSOCIATIVE CLASSIFICATION

The problem of AC is to discover a subset of rules with significant supports and high confidences. This subset is then used to build an automated classifier that could be used to predict the classes of previously unseen data. It should be noted that MinSupp and MinConf terms in ARM

(Association Rule Mining) are different than those defined in AC since classes are not considered in ARM, only itemsets occurrences are used for the computation of support and confidence.

The CBA algorithm[26] was one of the first AC(Associative Classification) algorithms that employed an Apriori candidate generation step to find the rules. Classification Based on Associations (CBA) was presented by (Liu et al., 1998) and it uses Apriori candidate generation method (Agrawal and Srikant, 1994) for the rule discovery step. CBA operates in three steps, where in step 1, it discretises continuous attributes before mining starts. In step 2, all frequent ruleitems which pass the MinSupp threshold are found, finally a subset of these that have high confidence are chosen to form the classifier in step3. Due to a problem of generating many rules for the dominant classes or few and sometime no rules for the minority classes, CBA (2) has introduced by (Liu *et al*. 1999), which uses multiple support thresholds for each class based on class frequency in the training data set. Experiment results have shown that CBA (2) outperforms CBA and C4.5 in terms of accuracy.

Classification based on Multiple Association Rules (CMAR)[23] adopts the FP-growth ARM algorithm (Han et al., 2000) for discovering the rules and constructs an FP-tree to mine large databases efficiently (Li et al., 2001). It consists of two phases, rule generation and classification. It adopts a FP- growth algorithm to scan the training data to find the complete set of rules that meet certain support and confidence thresholds. The frequent attributes found in the first scan are sorted in a descending order, i.e. F-list. Then it scans the training data set again to construct an FP-tree. For each tuple in the training data set, attribute values appearing in the F-list are extracted and sorted according to their ordering in the F-list. Experimental results have shown that CMAR is faster than CBA and more accurate than CBA and C4.5. The main drawback documented in CMAR is the need of large memory resources for its training phase.

Classification based on Predictive Association Rules (CPAR)[29] is a greedy method proposed by (Yin and Han, 2003). The algorithm inherits the basic idea of FOIL in rule generation (Cohen,1995) and integrates it with the features of AC.

Multi-class Classification based on Association Rule (MCAR)[28] is the first AC algorithm that used a vertical mining layout approach (Zaki et al.,1997) for finding rules. As it uses vertical layout, the rule discovery method is achieved through simple intersections of the itemsets Tid-lists, where a Tid-list contains the item's transaction identification numbers rather than their actual values. The MCAR algorithm consists of two main phases: rules generation and a classifier builder. In the first phase, the training data set is scanned once to discover the potential rules of size one, and then MCAR intersects the potential rules Tid-lists of size one to find potential rules of size two and so forth. In the second phase, the rules created are used to build a classifier by considering their effectiveness on the training data set. Potential rules that cover a certain number of training objects will be kept in the final classifier. Experimental results have shown that MCAR achieves 2-4% higher accuracy than C4.5, and CBA.

Multi-class, Multi-label Associative Classification (MMAC) [27] algorithm consists of three steps: rules generation, recursive learning and classification. It passes over the training data set in the first step to discover and generate a complete set of rules. Training instances that are associated with the produced rules are discarded. In the second step, MMAC proceeds to discover more rules that pass MinSupp and MinConf from the remaining unclassified instances, until no further potential rules can be found. Finally, rule sets derived during each iteration are merged to form a multi-label classifier that is then evaluated against test data. The distinguishing feature of MMAC is its ability to generate rules with multiple classes from data sets where each data objects is associated with just a single class. This provides decision makers with useful knowledge discarded by other current AC algorithms.

4. FINDING CLASS ASSOCIATION RULES

Apriori-based implementations are efficient but cannot generate all valid positive and negative ARs. In this section, we try to solve that problem without paying too high a price in terms of computational costs. Generating negative class association rules of the form ┐(= XY)=>C For simplicity, we also limit ourselves to support and confidence to determine the validity of ARs.

Algorithm:

1. **Generating negative class association rules of the form ┐I(= XY)=>C**
2. **Generate negative class association rules of the form ┐X → C**
3. **Generate negative class association rules of the form ┐X ┐*Y* → C**
4. **Generate negative class association rules of the form ┐XY → C**

4.1. Finding Positive class Association Rules XY=>C

1. AR ← φ;
2. S ← φ;
3. Find L $(P_1)_1$ i.e. Frequent 1-itemsets
4. L(P_1) ← L(P_1)
5. For k=2; L $(P_1)_{k-1}$≠Ø; k++
6. {
7. // Generating C_k
8. for each I_1,I_2 ε L$(P_1)_{k-1}$
9. If(I_1[1]=I_2[1]^..................I_1[k-2]=I_2[k-2]^I_1[k-1]<I_2[k-1])
10. C_k=Ck ∪ {{I_1 [1].......I_1 [k-2],I_1[k-1],I_2[k-1]}
11. end if
12. end for
13. // Pruning using Apriori property
14. for each (k-1)- subsets s of I ε C
15. If s ∉ L $(P_1)_{k-1}$
16. C_k=C_k – {I}
17. end if
18. end for
19. //Pruning using Support Count
20. Scan the database and find supp(I) for all I ε C_k
21. S=S ∪ {I with support count}
22. For each I in C_k
23. If supp (I) ≥ ms
24. L $(P_1)_k$=L $(P_1)_k$ ∪ {I}
25. end if
26. end for
27. L (P_1) =L(P_1) ∪ L$(P_1)_k$
28. }
29. end for
30. // Generating Positive Classification Rules of the form I(=XY) => c
31. for each I(=XY) ε L(P_1)
32. for each c ε C
33. If conf (I → c) ≥ mc
34. AR=AR ∪ {I → c}
35. end if
36. end for
37. end for

4.2. Generating negative class association rules of the form ┐I(= XY)=>C

1. for each I ε L(P1)
2. if 1-supp(I) ≥ ms

```
   L(P2)=L(P2) U I
end for
// Generating  Negative  Association Rules of the from ┐(XY) => c
for each I ε L(P2)
    for each  c ε C
      If conf(┐I→c)≥ mc
        AR=AR ∪ {┐I→c}
          end for
      end for
```

4.3. Generating negative class association Rules of the form I(=┐X┐Y) => C

```
C(P3 )2 = {¬{i1 }¬{i2 }|i1 , i2 ε L(P_1 )_1 , i1 ≠ i2 }
for {k = 2; C(P_3 )_k ≠ Ø; k + +} do
    for all I = ¬X¬Y ε  C(P_3 )_k do
      if supp(I) ≥ ms then
        insert I into L(P_3 )_k
      else
     for all i ∉ XY do
    // Generating Candidates
         Cand ={┐ (XU{i})┐Y, ┐X(┐Y U{i})}
            // Pruning Cand
              for each item in Cand
              If (X{i} is not in L(P_1) or  ┐X^1┐Y^1 is in  L(P3)  where X^1⊆X{i} and Y^1⊆Y)
              Cand= Cand – {XY {i}}
              C(P3)_k+1= C(P3)_k+1  U Cand
             if Cand ≠ ∅, XY {i} ∉ S and
          (!∃ I^1 ⊆ XY {i}) (supp(I^1) = 0) then
           insert XY {i} into S(P_3 )_k+1
           end if
            end for
          end if
       end for
    compute support of itemsets in S(P_3 )_k+1
    S = S ∪ S(P_3 )_k+1
    end for
    // Generating  Negative  class association  Rules of the from I(= ┐X┐Y) => C
    for each I ε L(P_3)
        for each  c ε C
          If conf(I→c)≥ mc
            AR=AR ∪ {I→c}
               If conf (I→┐c) ≥ mc
               AR=AR ∪ {I→┐c}
        end for
    end for
```

4.4. Generating negative class Association Rules of the form ┐XY=>C

```
C(P_4 )_1,1 = {¬{i_1 }{i_2 }|i_1 , i_2  ε L(P_1 )_1 , i_1 ≠ i_2 }
for {k = 1; C(P_4 )_k,1 = Ø; k + +} do
for {p = 1; C(P_4 )_k,p ≠ Ø; p + +} do
          for all I ε C(P_4 )_k,p do
                    if supp(I) ≥ ms then
                  insert I into L(P_4 )_k,p
end if
```

8. end for
9. //Generating Candidates
10. // I_1 and I_2 are joinable if $I1_1 \neq I_2$, I_1 .negative = I_2 .negative, I_1 .positive and //I_2.positive share the same k – 1 items, and I_1 .positive U I_2 .positive ε $L(P_1)_{p+1}$
11. for all joinable I_1 , I_2 ε $L(P_4)_{k,p}$ do
12. X = I_1 .negative, Y = I_1 .positive U I_2 .positive
13. I = ¬XY
14. if $(!\exists X^1 \subset X)(supp(\neg X^1\ Y) \geq ms)$ and $(!\exists Y^1 \subset Y)(supp(\neg XY^1) < ms)$ then insert I into $C(P_4)_{k,p+1}$
15. if $XY \notin S$ and $!\exists I^1 \subset XY$, $supp(I^1) = 0$ then
16. insert XY into $S(P_4)_{k,p+1}$
17. end if
18. end if
19. end for
20. compute support of itemsets in $S(P_4)_{k,p+1}$
21. $S = S \cup S(P_4)_{k,p+1}$
22. end for
23. for all X ε $L(P_1)_{k+1}$, i ε $L(P_1)_1$ do
24. if $(!\exists X^1 \subset X)(\neg X^1 \{i\}$ ε $L(P_4))$ then $C(P_4)_{k+1,1} = C(P_4)_{k+1,1} \cup \neg X\{i$
25. end if
26. end for
27. end for
28. // Generating Negative Association Rules of the from ┐XY=> C
29. for each I ε $L(P_4)$
30. for each c ε C
31. If conf(I→c)≥ mc
32. AR=AR ∪ {I→c}
33. If conf(I→┐c)≥ mc
34. AR=AR ∪{I->┐c}
35. end for
36. end for

5. ASSOCIATIVE CLASSIFIER

The set of rules that were generated as discussed in the previous section represent the actual classifier. This categorizer is used to predict to which classes new objects are attached. Given a new object, the classification process searches in this set of rules for those classes that are relevant to the object presented for classification. The set of positive and negative rules discovered as explained in the previous section are ordered by confidence and support. This sorted set of rules represents the associative classifier. This subsection discusses the approach for labeling new objects based on the set of association rules that forms the classifier

Algorithm: CPNAR (Classification based on **P**ositive and **N**egative **A**ssociation **R**ules)

Input: A new object to be classified o;
The associative classifier (AC);
The confidence margin T;
Output: Category attached to the new object

Method:
1. S←φ /* set of rules that match o*/
2. for each r in AC /* the sorted set of rules */
3. If (r⊂o) {count++}
4. S←S U r
5. If(count==1)
6. fr.conf←r.conf /* keep the first rule confidence*/

7. $S \leftarrow S \cup r$
8. else if(r.conf>fr.conf- τ)
9. $S \leftarrow S \cup r$
10. else break
11. Divide S in subsets by category:$S_1, S_2, \ldots S_n$
12. for each subset $S_1, S_2, \ldots S_n$
13. Sum/substract the confidences of rules and divide by the number of rules in S_k
14. $Score_i = \Sigma r.conf/\#rules$
15. Put the new object in the class that has the highest confidence score
16. $o \rightarrow c_i$, with $score_i = \max\{ score_1, \ldots score_n\}$

In the above algorithm (Classification of a new object), a set of applicable rules is selected in the lines 1-8. The set of applicable rules is selected within a confidence margin. The interval of selected rules is between the confidence of the first rule and this confidence minus the confidence margin as checked in line 7. The prediction process is starting at line 10. The applicable set of rules is divided according to the classes in line 10. In lines 11-12 the groups are ordered according to the average confidence per class. In line 13 the classification is made by assigning to the new object the class that has the highest score.

6. EXPERIMENTAL RESULTS

The implementation of our algorithm is a java program. The experiments have been performed using datasets downloaded from UCI machine learning repository. To run the experiments, we have used ten-fold cross validation test to compute the accuracy of the classifier. To discretize the continuous attributes, we have adopted the technique used in CBA. All the experiments are performed on a 600 MHz Pentium PC with 128MB main memory running Microsoft XP. From the table 2, CPNAR algorithm has performed well for Heart, Iris and Zoo datasets when compared to C4.5,CBA,CMAR and CPAR.

DATASET	#ATTS	#CLS	#REC	#Rules Generated
BREST	10	2	699	478
HEART	13	2	270	209
HEPATITIS	19	2	155	87
IRIS	4	3	150	123
ZOO	16	7	101	68

TABLE 1: No. of CARs generated by our algorithm on various UCI ML datasets

DATASET	C4.5	CBA	CMAR	CPAR	CPNAR
BREST	95.0	96.3	**96.4**	96.0	96.6
HEART	80.8	81.9	82.2	82.6	**83.0**
HEPATITIS	80.6	81.8	80.5	**82.6**	82.3
IRIS	95.3	94.7	94	94.7	**95.6**
ZOO	92.2	96.8	97.1	95.1	**97.5**

TABLE 2: Accuracies of Various Classifiers on UCI ML datasets

7. CONCLUSION AND FUTURE WORK

We proposed an algorithm that integrates classification and association rule generation. It mines both positive and negative class association rules. Our method generates positive and negative class association rules with existing support-confidence framework. We conducted experiments on UCI datasets. In future we wish to improve accuracy of our algorithm and then we conduct experiments on some more datasets and compare the performance with other related algorithms.

8. REFERENCES

[1] B.Ramasubbareddy, A.Govardhan, and A.Ramamohanreddy. Adaptive approaches in mining negative association rules. In Intl. conference on ITFRWP-09, India Dec-2009

[2] B.Ramasubbareddy, A.Govardhan, and A.Ramamohanreddy. Mining Positive and Negative Association Rules, IEEE ICSE 2010, Hefeai, China, August 2010.

[3] R. Agrawal and R. Srikant. *Fast algorithms for mining associationrules.* In VLDB, Chile, September 1994.

[4] J. Han, J. Pei, and Y. Yin. *Mining frequent patterns* without candidate generation. In SIGMOD, dallas, Texas, 2000.

[5] C. Blakeand C. Merz. UCI repository of machine learning databases.

[6] S.Brin, R. Motwani, and C.Silverstein. *Beyond market* baskets: Generalizing association rules to correlations. In ACM SIGMOD, Tucson, Arizona, 1997.

[7] D. Thiruvady and G. Webb. *Mining negative association rules using grd*. In PAKDD, Sydney, Australia, 2004

[8] Goethals, B., Zaki, M., eds.: *FIMI'03: Workshop on* Frequent Itemset Mining Implementations. Volume 90 of CEUR Workshop Proceedings series. (2003) http://CEUR-WS.org/Vol-90/.

[9] Teng, W., Hsieh, M., Chen, M.: *On the mining of* substitution rules for statistically dependent items. In: Proc. of ICDM. (2002) 442–449

[10] Tan, P., Kumar, V.: Interestingness measures for association patterns: A perspective.In: Proc. of Workshop on Postprocessing in Machine Learning and Data Mining. (2000)

[11] Gourab Kundu, Md. Monirul Islam, Sirajum Munir, Md. Faizul Bari ACN: An Associative Classifier with *Negative Rules* 11th IEEE International Conference on Computational Science and Engineering, 2008.

[12] Brin,S., Motwani,R. and Silverstein,C., " *Beyond Market* Baskets: Generalizing Association Rules to *Correlations,*" Proc. ACM SIGMOD Conf., pp.265-276, May 1997.

[13] Chris Cornelis, peng Yan, Xing Zhang, Guoqing Chen: Mining Positive and Negative Association Rules from Large Databases , IEEE conference 2006.

[14] M.L. Antonie and O.R. Za¨ıane, "*Mining Positive and* Negative Association Rules: an Approach for Confined *Rules"*, Proc. Intl. Conf. on Principles and Practice of Knowledge Discovery in Databases, 2004, pp 27–38.

[15] Savasere, A., Omiecinski,E., Navathe, S.: *Mining for* Strong negative associations in a large data base of *customer transactions.* In: Proc. of ICDE. (1998) 494- 502..

[16] Wu, X., Zhang, C., Zhang, S.: efficient m*ining both positive and negative association rules*. ACM Transactions on Information Systems, Vol. 22, No.3, July 2004,Pages 381-405.

[17] Wu, X., Zhang, C., Zhang, S.: *Mining both positive and* negative association rules.In: Proc. of ICML. (2002) 658–665

[18] Yuan,X., Buckles, B.,Yuan, Z.,Zhang, J.:*Mining* Negative Association Rules. In: Proc. of ISCC. (2002) 623-629.

[19] Honglei Zhu, Zhigang Xu: *An Effective Algorithm for* Mining Positive and Negative Association Rules. International Conference on Computer Science and Software Engineering 2008.

[20] Pradip Kumar Bala:A Technique for Mining Negative *Association Rules .* Proceedings of the 2nd Bangalore Annual Compute Conference (2009).

[21] Data Mining: Concepts and Techniques Jiawei Han, Micheline Kamber

[22] Quinlan, J. 1993 C4.5: Programs for Machine Learning. San Mateo, CA: Morgan Kaufmann

[23] Li, W., Han, J. & Pei, J. 2001 CMAR: Accurate and efficient classification based on multiple-class association rule. In Proceedings of the International Conference on Data Mining (ICDM'01), San Jose, CA, pp. 369–376

[24] Dong, G., Zhang, X., Wong, L. & Li, J. 1999 CAEP: Classification by aggregating emerging patterns. In Proceedings of the 2nd Imitational Conference on Discovery Science. Tokyo, Japan: Springer Verlag, pp. 30–42.

[25] Antonie, M. & Zaïane, O. 2004 An associative classifier based on positive and negative rules. In Proceedings of the 9th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery. Paris, France: ACM Press, pp. 64–69

[26] B. Liu, W. Hsu, and Y. Ma. Integrating classification and association rule mining. In *ACM Int. Conf. on Knowledge Discovery and Data Mining (SIGKDD'98)*, pages 80–86, New York City, NY, August 1998.

[27] Thabtah, F., Cowling, P. & Peng, Y. 2004 MMAC: A new multi-class, multi-label associative classification approach. In Proceedings of the 4th IEEE International Conference on Data Mining (ICDM'04), Brighton, UK, pp. 217–224.

[28] Thabtah, F., Cowling, P. & Peng, Y. 2005 MCAR: Multi-class Classification based on Association Rule approach. In Proceeding of the 3rd IEEE International Conference on Computer Systems and Applications,Cairo, Egypt, pp. 1–7.

[29] Yin, X. & Han, J. 2003 CPAR: Classification based on predictive association rule. In Proceedings of the SIAM International Conference on Data Mining. San Francisco, CA: SIAM Press, pp. 369–376. B. Liu, W. Hsu, &Y. Ma, "*Integrating classification and association rule mining*", Proceeding of KDD'98, 1998, pp. 80-86.

[30] B.Ramasubbareddy, A.Govardhan, A.Ramamohanreddy, An Approach for Mining Positive and Negative Association Rules, Second International Joint Journal Conference in Computer, Electronics and Electrical, CEE 2010

[31] B.Ramasubbareddy,A.Govardhan,A.Ramamohanreddy, Mining Indirect Association between Itemsets, proceedings of Intl conference on Advances in Information Technology and Mobile Communication-AIM-2011 published by Springer LNCS , April 21-22, 2011, Nagapur, Maharastra, India

[32] B.Ramasubbareddy, A.Govardhan, and A.Ramamohanreddy Mining Indirect Positive and Negative Association Rules, Intl Conference on Advances in Computing and Communications, July 22-24 2011, Kochi, India