© 2016 IJSRSET | Volume 2 | Issue 5 | Print ISSN: 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

An Improved Association Rule Mining with Frquent Itemset Relationship Technique

Prof.Neeraj Shukla, Arpita Sen

Department of Computer Science & Engineering, Gyan Ganga College of Technology Jabalpur, Madhya Pradesh, India

ABSTRACT

Construction also, improvement of classifier that work with more precision and perform productively for vast database is one of the key errand of information mining methods [l7] [18]. Besides preparing dataset over and over produces huge measure of principles. It's exceptionally difficult to store, recover, prune, and sort an enormous number of standards capably before applying to a classifier [1]. In such circumstance FP is the best decision yet issue with this methodology is that it produces repetitive FP Tree. A Frequent example tree (FP-tree) is a sort of prefix tree [3] that permits the identification of repetitive (continuous) thing set restrictive of the competitor thing set era [14]. It is expected to recover the blemish of existing mining strategies. FP-Trees seeks after the gap and overcomes strategy. In this paper we have embrace the same thought of creator [17] to manage vast database. For this we have incorporated a positive and negative tenet mining idea with regular example (FP) of characterization. Our technique performs well and creates special tenets without uncertainty.

Keywords : Association, FP, FP-Tree, Nagtive, Positive

I. INTRODUCTION

Mining utilizing Association rules find engaging connections or relationship among the information things sets from colossal measure of information [4]. For this affiliation utilizes different strategies like Apriori and continuous example rules, despite the fact that Apriori utilize cut-innovation while creating thing sets, it look at the entire database amid checking of the exchange database inevitably. This subsequent filtering pace is slowly diminished as the information size is developing [4].

Second understood calculation is Frequent Pattern (FP) development calculation it takes up separation and-overcome approach. FP processes the incessant things and structures in a tree of regular example.

In correlation with Apriori calculation FP is much predominant if there should be an occurrence of productivity [13]. In any case, issue with conventional FP is that it delivers an enormous number of contingent FP trees [3].

Development and advancement of classifier that work with more exactness and perform productively for huge database is one of the key errand of information mining methods [l7] [18]. Also preparing dataset over and over produces huge measure of standards. It's extremely difficult to store, recover, prune, and sort an enormous number of guidelines capably before applying to a classifier [1]. For dispose of such issues Author of [17] proposed another technique in light of positive and negative idea of affiliation principle mining. Creators contend that the standard strategies of grouping in light of the positive affiliation leads and disregards the estimation of negative affiliation rules.

In this paper we have get the same considered maker [17] to oversee broad database. For this we have facilitated a positive and negative rule mining thought with perpetual example (FP) of grouping. Our technique performs well and creates special principles without equivocalness.

Rest of papers are sorted out as takes after, area two understanding the foundation subtle elements of the affiliation information mining method furthermore investigate the possibility of FP and positive and negative hypothesis. Segment 3 talks about the past works in same field. Area 4 talks about the proposed technique and calculation received. Segment 5 introduces the outcomes acquired by the proposed technique lastly area 6 finishes up the paper.

II. METHODS AND MATERIAL

1. **Wireless sensor** Backgrounds & Related terminology

A. Association

Association rule was proposed by Rakesh Agrawal [1]; its uses the "if-then" rules to generate extracted information into the form transaction statements [3]. Such rules have been created from the dataset and it obtains with the help of support and confidence of apiece rule that illustrate the rate (frequency) of occurrence of a given rule.

According to the Author of [2] Association mining may be can he stated as follows: Let I = (i1,i2…in) be a set of items. Let D = (T1, T2…Tj,…Tm) the task-relevant data, be a set of transactions in a database, where each transaction Tj(j=1, 2, ⋯ , m) such that Tj ⊆I . Each transaction is assigned an identifier, called TID (Transaction id). Let A be a set of items, a transaction T is said to contain A if and only if A⊆I. An association rule is an implication of the form A→B where A⊆I , B⊆I and
A∩B=∅. The rule A→B holds in the transaction set D with support s, where s is the percentage of transactions in D that contain A B (i.e., both A and B). This is taken to be the probability P(A B). The rule has confidence c in the transaction set D if c is the percentage of transactions in D containing A that also contain B. This is taken to be the conditional probability, P (B|A). That is,
confidence(A->B)= P (B|A)=support(A¬B) ／ support(A)=c , support(A->B)=P(A¬B)=s.
The popular association rules Mining is to mine strong association rules that satisfy the user specified both minimum support threshold and confidence threshold. That is, minconfidence and minsupport. If support(X) ≥minsupport, X is frequent item sets. Frequent k-itemsets is always marked as LK. If support (A->B) ≥minsupport and confidence (A->B)≥minconfidence A->B is strong correlation. Several Theorems are introduced as follows:

(i)If A⊆B, support(A)≥support(B).
(ii)If A⊆B and A is non-frequent itemset, then B is non-frequent itemset.
(iii) If A⊆B and B is frequent itemset, then A is frequent itemset.

B. Frequent Pattern (FP) Tree

A Successive example tree (FP-tree) is a kind of prefix tree [3] that permits the location of repetitive (incessant) thing set restrictive of the hopeful thing set era [14]. It is expected to recover the defect of existing mining techniques. FP –Trees seek after the gap and vanquishes strategy. The foundation of the FP-tree is tag as "Invalid" worth. Childs of the roots are the arrangement of thing of information. Routinely a FP tree contains three fields-Item name, hub connection and tally.

To maintain a strategic distance from various restrictive FP-trees amid mining of information creator of [3] has proposed another affiliation guideline mining procedure utilizing enhanced regular example tree (FP-tree) utilizing table idea conjunction with a mining continuous thing set (MFI) technique to take out the excess contingent FP tree.

Positive and Negative FP Rule Mining

Author of [15] cleverly explain the concept of positive and negative association rules. According to the [15] two indicators are used to decide the positive and negative of the measure:
1) Firstly find out the correlation according to the value of
corrP,Q=s(P∪Q)/s(P)s(Q),which is used to delete the contradictory association rules emerged in mining process.
There are three measurements possible of corrP, Q [16]:
• If corrP, Q>1, Then P and Q are related;
• If corrP, Q=1, Then P and Q are independent of each other;
• If corrP, Q<1, Then P and Q negative correlation;
2) Support and confidence is the positive and negative association rules in two important indicators of the measure.
The support given by the user to meet the minimum support (minsupport) a collection of itemsets called frequent itemsets, association rules mining to find

frequent itemsets is concentrating on the needs of the user to set the minimum confidence level (minconf) association rules.

Negative association rules contains itemset does not exist (non-existing-items, for example ¬P, ¬Q), Direct calculation of their support and confidence level more difficult.

2. Literature Survey

Data mining is used to deal with size of data stored in the database, to extract the desired information and knowledge [3]. Data mining has various technique to perform data extraction association technique is the most effective data mining technique among them. It discover hidden or desired pattern among the large amount of data. It is responsible to find correlation relationships among different data attributes in a large set of items in a database. Since its introduction, this method has gained a lot of attention. Author of [3] has analyzed that an association analysis [1] [5] [6] [7] is the discovery of hidden pattern or clause that occur repeatedly mutually in a supplied data set. Association rule finds relations and connection among data and data sets given.

An association rule [1] [5] [8] [9] is a law which necessitate certain relationship with the objects or items. Such association's rules are calculated from the data with help of the concept of probability.

Association mining using Apriori algorithm perform better but in case of large database it performs slow because it has to scan the full database each time while scanning the transaction as author of [4] surveyed.

Author of [3] has surveyed and conclude with the help of previous research in data mining using association rules has found that all the previously proposed algorithm like - Apriori [10], DHP [11], and FP growth [12] .

Apriori [6] employ a bottom-up breadth-first approach to discover the huge item set. The problem with this algorithm is that it cannot be applied directly to mine complex data [3]. Second well-known algorithm is Frequent Pattern (FP) growth algorithm it takes up divide-and-conquer approach. FP computes the frequent items and forms in a tree of frequent-pattern. In comparison with Apriori algorithm FP is much superior in case of efficiency [13]. But problem with traditional FP is that it produces a huge number of conditional FP trees [3].

3. Improved Association Rule Mining with Frequent Itemset Relationship Technique

Existing work in light of Apriori calculation for finding continuous example to produce affiliation manages then apply class name affiliation rules where this work utilizes FP tree with development for finding regular example to create affiliation rules. Apriori calculation sets aside more opportunity for vast information set where FP development is time proficient to discover continuous example in exchange.

In this paper we have propose another measurement into the information mining strategy. For this we have coordinated the idea of positive and negative affiliation rules into the regular example (FP) strategy. Negative and positive standards works preferred for over conventional affiliation guideline mining and FP shrewdly works in expansive database. Our proposed strategy fill in as takes after-Positive and Negative class association rules based on FP tree.

This algorithm has two stages: rule generation and classification. In the first stage: the algorithm calculate the whole set of positive and negative class association rules such that sup(R) support and conf(R) confidence given thresholds. Furthermore, the algorithm prunes some contradictory rules and only selects a subset of high quality rules for classification.

In the second stage: classification, for a given data object, the algorithm extracts a subset of rules fund in the first stage matching the data object and predicts the class label of the data object by analyzing this subset of rules.

1) Generating Rules

To find rules for classification, the algorithm first mines the training dataset to find the complete set of rules passing certain support and confidence thresholds. This is a typical frequent pattern or association rule mining task. The algorithm adopts FP Growth method to fmd frequent itemset. FP Growth method is a frequent itemset mining algorithm which is fast. The algorithm also uses the correlation between itemsets to find positive and negative class association rules. The correlation between itemsets can be defined as:

$corr(X,Y)=(sup(X \cup Y))/(sup(X)sup(Y))$

X and Y are itemsets.

When corr(X, Y)>1, X and Y have positive correlation.

When corr(X, Y)=1, X and Yare independent.

When corr(X, Y)<1, X and Y have negative correlation.

Also when corr(X, Y)>1, we can deduce that corr(X, -Y)<1 and corr(-X,Y)<1.

So, we can use the correlation between itemset X and class label ci to judge the class association rules.

When corr(X, ci)> 1, we can deduce that there exists the positive class association rule X → ci

When corr(X, ci)> 1, we can deduce that there exists the negative class association rule X→-ci

So, the first step is to generate all the frequent itemsets by making multiple passes over the data. In the first pass, it counts the support of individual itemsets and determines whether it is frequent. In each subsequent pass, it starts with the seed set of itemsets found to be frequent in the previous pass. It uses this seed set to generate new possibly frequent itemsets, called candidate itemsets. The actual supports for these candidate itemsets are calculated during the pass over the data. At the end of the pass, it determines which of the candidate itemsets are actually frequent.

The algorithm of generating frequent itemsets is shown as follow:

a) Definition FP-tree: A frequent-pattern tree (or FP-tree) is a tree structure defined below.

- It consists of one root labeled as "null", a set of item-prefix subtrees as the children of the root, and a frequent-item-header table.
- Each node in the item-prefix subtree consists of three fields: item-name, count, and ode-link, where item-name registers which item this node represents, count registers the number of transactions represented by the portion of the path reaching this node, and node-link links to the next node in the FP-tree carrying the same item-name, or null if there is none.
- Each entry in the frequent-item-header table consists of two fields, (1) item-name and (2) head of node-link (a pointer pointing to the first node in the FP-tree carrying the item-name).

Based on this definition, we have the following FP-tree construction algorithm.

b) Algorithm for FP-tree construction

Input: A transaction database DB and a minimum support threshold ξ .

Output: FP-tree, the frequent-pattern tree of DB.

Method: The FP-tree is constructed as follows.

1. Scan the transaction database DB once. Collect F, the set of frequent items, and the support of each frequent item. Sort F in support-descending order as FList, the list of frequent items.
2. Create the root of an FP-tree, T, and label it as "null". For each transaction Trans in B do the following-

- Select the frequent items in Trans and sort them according to the order of FList. Let the sorted frequent-item list in Trans be [p | P], where p is the first element and P is the remaining list. Call insert tree ([p | P], T).
- The function insert tree([p | P], T) is performed as follows. If T has a child N such that N.item-name = p.item-name, then increment N's count by 1; else create a new node N, with its count initialized to 1, its parent link linked to T , and its node-link linked to the nodes with the same item-name via the node-link structure. If P is nonempty, call insert tree (P, N) recursively.
- Then, the next step is to generate positive and negative class association rules. It firstly finds the rules contained in F which satisfy min_sup and min_conf threshold. Then, it will determined the rules whether belong to the set of positve class correlation rules P_AR or the set of negative class correlation rules N_AR.

The algorithm of generating positive and negative class association rules is shown as follow:

c) Algorithm for generating positive and negative class association rules

Input: training dataset T, min_sup, min_conf

Output: P_AR, N_AR

(I)P_AR=NULL, N_AR=NULL;

```
2)  for (any frequent itemset X in F and Ci in C)
   {
    if (sup(X→ci)>min_sup and conf(X→ ci)> min_conf)
    if( corr(X, ci > 1)
     {
          P_AR= P_AR U {X→ - ci;};
        }
   else if corr(X, ci <I
    {
      N_AR= N_AR U {X→ - ci;};
     }
}
```

3) ReturnP_AR and N_AR;

In this algorithm, we use FP Growth method generates the set of frequent itemsets F, In F, there are some itemsets passing certain support and confidence thresholds. And the correlation between itemsets and class labels is used as an important criterion to judge whether or not the correlation rule is positve. Lastly, P_AR and N_AR are returned.

C. Classification

After P_AR and N_AR are selected for classification, the algorithm is ready to classify new objects. Given a new data object, the algorithm collects the subset of rules matching the new object. In this section, we discuss how to determine the class label based on the subset of rules.

First, the algorithm finds all the rules matching the new object, generates PL set which includes all the positive rules from P _ AR and sorts the itemset by descending support values. The algorithm also generates NL set which includes all the negative rules from N_AR and sort the itemset by descending support values. Second, the algorithm will compare the positive rules in PL with the negative rules in NL and decides the class label of the data object.

The algorithm of classification is shown as follow:

D. Algorithm for Classification

```
Input: data object, P _AR, N_AR
Output: the class label of data object Cd
(1) PL=Sort(P_AR); NL=Sort{N_AR); i=j=I;
(2)pJule=GetElem(pL, i); nJule=GetElem(NL,j);
(3)while Ci<=PL_Length and j<=NL_Length)
    {
        if(RuleCompare(p _role, n _role))
        {
        if(P _role>n _role)
        {
        }
        Cd = the label of p_role;
        Break;
        if(P _role=n _role)
        {
        }
        Cd = the label of p _role;
        break;
        if(P _role<n _role)
        {
        j+ + ;
        }
        }
        if(!RuleCompare(pJule, nJule))
        {
        if(P Jule>n Jule)
        {
        Cd = the label ofpJule;
        break;
        }
        }
        if(P _ rule=n_ rule)
        {
        i+ + ;
        j+ + ;
        }
        if(P _ rule<n _rule)
        { i+ + ;
        }
(4)return Cd;
```

In the algorithm of classification, the function Sort(P_AR) returns PL and the itemsets in PL are sorted by descending support values, the function GetElem(pL, i) returns first I rule in the set of PL. Also, we can deduce the returns of the function of Sort{N_AR) and GetElem{NL,j).

III. RESULTS AND PERFORMANCE MEASUREMENT

Proposed enhanced FP with positive and negative system has been implement using java technologies. Following results have been measured by the system.
Settings

File name = data.num
Support (default 20%) = 20.0
Confidence (default 80%) = 80.0
Reading input file: data.num
Number of records = 95
Number of columns = 38
Min support = 19.0 (records)
Generation time = 0.0 seconds (0.0 mins)
FP tree storage = 2192 (bytes)
FP tree updates = 694
FP tree nodes = 97

FP Tree

(1) 9:90 (ref to null)
(1.1.1.1.1) 1:72 (ref to 1:4)
(1.1.1.1.1.1) 32:65 (ref to 32:3)
And so on…………

GENERATING ARs:
Generation time = 0.17 seconds (0.0 mins)
T-tree Storage = 8824 (Bytes)
Number of frequent sets = 626

[1] {9} = 90
[2] {19} = 90
[3] {19 9} = 85
[4] {23} = 90
And so on…….(Approximate 624 generated)

ASSOCIATION RULES

(1) {1 32 5} -> {19} 100.0%
(2) {9 1 32 5} -> {19} 100.0%
.
.
.
(102) {9 23 32 14 37} -> {27} 100.0%
(103) {9 27 32 14 37} -> {23} 100.0%
(104) {9 32 14 37} -> {23 27} 100.0%
And so on……(Approximate 7855 generated)

Possitive Class Itemsets RULES

{9 27 1 32 14} -> {19}
{9 27 1 32 14} -> {23}
{19 27 1 32 14} -> {23}
{9 19 27 1 32 14} -> {23}
{19 23 27 1 32 14} -> {9}
And so on…….

Negative Class Itemsets RULES

{9 14 37} -> ~ {23 27}
{9 19 23 14 37} -> ~ {27}
{9 19 14 37} -> ~ {23 27}
{9 1 14 37} -> ~ {23}
{9 19 1 14 37} -> ~ {23}
And so on……

The result shows that the proposed system works more efficiently than exiting positive and negative using Apriori technique. We have evaluated that it can handle very large data set and able to mine efficiently. A current experiment shows that it can handle data 129941 KB of data. This statistics is chosen by us. Even our system can handle and generate more mined data.

IV. CONCLUSION

In this paper we have proposed another half and half way to deal with for information mining process. Information mining is the momentum center of exploration since a decade ago because of gigantic measure of information and data in cutting edge. Affiliation is the intriguing issue among different information mining procedure. In this article we have proposed a half and half way to deal with manage huge size information. Proposed framework is the upgrade of Frequent example (FP) procedure of relationship with positive and negative combination on it. Conventional FP strategy performs well yet produces excess trees coming about that effectiveness corrupts. To accomplish better effectiveness in affiliation mining positive and negative tenets era out. Same idea has been connected in the proposed technique. Results shoes that propsed strategies perform well and can deal with expansive size of information set.

V. REFERENCES

[1]. Agrawal R,Imielinski T,Swami A, “Mining Association Rules between Sets of Items in Large Databases, ”In:Proc of the ACM SIGMOD International conference on Management of Data,Washington DC,1993,pp,207-216.

[2]. QI Zhenyu, XU Jing, GONG Dawei and TIAN He “Traversing Model Design Based on Strong-association Rule for Web Application Vulnerability Detection”, IEEE, International Conference on Computer Engineering and Technology, 2009.

[3]. A.B.M.Rezbaul Islam and Tae-Sun Chung “An Improved Frequent Pattern Tree Based Association Rule Mining Technique”, IEEE, International Conference on Information Science and Applications (ICISA), 2011

[4]. LUO XianWen and WANG WeiQing “Improved Algorithms Research for Association Rule Based on Matrix”, IEEE, International Conference on Intelligent Computing and Cognitive Informatics, 2010.

[5]. R Srikant, Qouc Vu and R Agrawal “Mining Association Rules with Item Constrains”. IBM Research Centre, San Jose, CA 95120, USA.

[6]. R Agrawal and R Srikant “Fast Algorithm for Mining Association Rules”. Proceedings of VLDB conference pp 487 – 449, Santigo, Chile, 1994.

[7]. J. Han and M. Kamber.Data Mining: Concepts and Techniques. Morgan Kaufman, San Francisco, CA, 2001.

[8]. Ashok Savasere, E. Omiecinski and ShamkantNavathe“An Efficient Algorithm for Mining Association Rules in Large Databases”, Proceedings of the 21st VLDB conference Zurich, Swizerland, 1995.

[9]. Arun K Pujai “Data Mining techniques”, UniversityPress (India) Pvt. Ltd., 2001.

[10]. J. S. Park, M.-S. Chen and P. S. Yu, “An effective Hash-Based Algorithm for Mining Association Rules”, Proceedings of the ACM SIGMOD, San Jose, CA, May1995, pp. 175-186.

[11]. S. Brin, R. Motwani, J. D. Ullman and S. Tsur, “Dynamic Item set Counting and Implication Rules for Market Basket Data”, Proceedings of the ACM SIGMOD, Tucson, AZ, May 1997, pp. 255-264.

[12]. J. Han, J. Pei, and Y. Yin, “Mining Frequent Patterns without Candidate Generation”, Proceedings of the ACM SIGMOD, Dallas, TX, May 2000, pp. 1-12.

[13]. Qin Ding and gnanasekaran Sundaraj “Association rule mining from XML data”, Proceedings of the conference on data mining.DMIN’06.

[14]. C. Silverstein, S. Brin, and R. Motwani, “Beyond Market Baskets: Generalizing Association Rules to Dependence Rules,” Data Mining and Knowledge Discovery, 2(1), 1998, pp 39–68.

[15]. Yanguang Shen, Jie Liu and Jing Shen “The Further Development of Weka Base on Positive and Negative Association Rules”, IEEE, International Conference on Intelligent Computation Technology and Automation (ICICTA), 2010.

[16]. Yanguang Shen,Jie Liu,Fangping Li.Application Research on Positive and Negative Association Rules Oriented Software Defects,2009 International Conference on Computational Intelligence and Software Engineering(CISE 2009)[C].Wuhan,China,December 11-13,2009.

[17]. LuoJunwei and Luo Huimin “Algorithm for Classification Based on Positive and Negative Class Association Rules”, 3rd IEEE International Conference on Computer Science and Information Technology (ICCSIT), 2010.

[18]. Wenmin Li, Jiawei Han and Jian Pei, "CMAR: Accurate and Efficient Classification Based on Multiple Class- Association Rules," Proceedings of the 2001 IEEE International Conference on Data Mining, IEEE Press, Dec. 2001, pp. 123-131.