3rd National Conference on Innovative Research Trends in Computer Science and Technology (NCIRCST 2018) ISSN: 2454-4248
Volume: 4 Issue: 3 155 – 159

Mining Class Association Rule with Itemset Constraint Using Parallel Computing

Vinothini.S, Vasanthaparkavi.P, Shubathra.S.
Student, student, student
Mr.T.Kumaravel Assistant Professor
Department of Computer Science and Engineering Kongu Engineering College
Perundurai-638060 Tamil Nadu
India

Abstract: Numerous fast algorithms for mining class-association rules (CARs) have beendeveloped recently. End-users are ofteninterested in a subset of class-associationrules. The nave strategy is to apply suchitem constraints into the post-processingstep. However, such approaches requiremuch effort and time. First, we built alattice structure to store all the frequentitemset constraints. Second, we develop atheorem for quickly pruning infrequentnodes. Final, an efficient algorithm formining CARs with item constraints isproposed. In this paper, we apply a parallelcomputing to solve the problem of CARs tospeed up the process on very large datasetsExperiments show that th proposed algorithm outperforms the mining time and memory usage.

*****

1. Introduction:

Association rule learning is a rule-based machine learning method for discovering interesting relations between variables in large databases**.** Association rule mining is a procedure which is meant to find frequent patterns, correlations, associations, or causal structures from data sets found in various kinds of databases such as relational databases, transactional databases, and other forms of data repositories .A class association rule is a special case of association rule where the rule consequent contains only a class label. In recent years, numerous approaches have been proposed to solve the problem of finding a complete set of CARs, that should satisfy user-specified minimum support and minimum confidence thresholds. CAR mining has been applied in many domains, such as healthcare, hotel and social security [1].Mining CARs with the itemset constraint has also been demonstrated in the public health domain. Therefore, an ongoing problem is mining CARs with the itemset constraint. This task can be accomplished by checking the rules obtained with the constraint in the pre- processing or post-processing step. Firstly, we built the lattice structure to store the frequent itemset in the dataset. In this paper,we propose a parallel computing to speed up the process on very large datasets. The parallel computing are considered as a better solution for big data mining.

2. Definitions:

Definiton1: A class association rule set (CARs) is a subset of association rules with classes specified as their consequences.

Definiton2: The support Supp(R),of rule is defined as , an indication of how frequently the items appear in the database

Definition3: The actual occurrence *ActOcc(R)* of rule *R* in *D* is the number of records of *D* that match *R*'s antecedent.

Definition 4: The confidence is defined as an association rule is a percentage value that shows how frequently the rule head occurs among all the groups containing the rule body. It is denoted as,

$$Conf=Supp(R)/ActOcc(R)$$

3. Problem statement:

Given a dataset *D*, an itemset constraint β, a minimum support threshold δ, and a minimum confidence threshold σ, the problem of mining

CARs with the itemset constraint is to find all CARs satisfying three constraints: the support constraint, the confidence constraint, and the itemset constraint[1].

4. Related studies:

Mining association rule

There are three main strategies to solve the problem of mining association rules. The first strategies is the post –processing methods, mine the frequent itemset using the Apriori or FP-Growth. The second strategies is the pre-processing methods[1] to filters out records which do not contain the constrained itemset in the pre-processing step. The third group is the constrained itemset filtering to generate only frequent itemset which satisfy the constraint.

The implementation of these three strategies with the itemset constraint consist of two main phases. In the first phase, frequent itemset are mined from the dataset. In the second phase, the set of association rule with the itemst constraint is generated. The problem of mining CARs with the itemset constraint thus requires a different strategy. In the next section, we review existing approaches for mining CARs with the itemset constraint.[1]

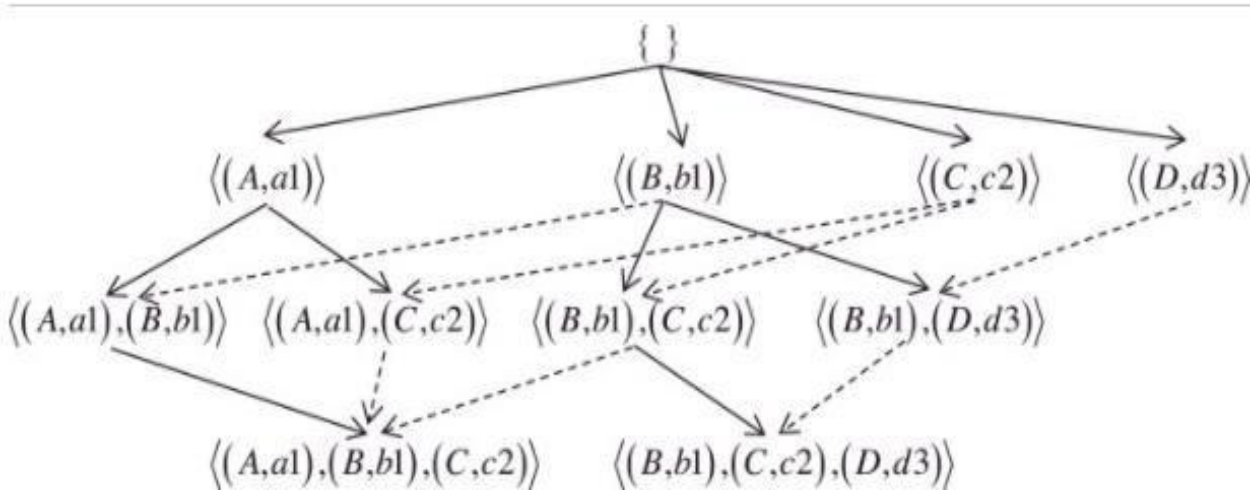
Mining class association rule

There are two basic approaches to discover CARs with the itemset constraint: post-processing and pre-processing. The post-processing approach uses the all CAR mining algorithm for improving the mining time and memory usage such as CMAR[4],CBA[3],MAC[5],there are some methods CMAR adopted an FP-tree-based approach for mining CARs .CMAR scans the dataset two times and uses a complex data structure to mine CARs. ECR-CARM proposed a new data structure (ECR-tree) to store itemsets. It only scans dataset one time[2]. The main advantage of pre-processing strategy is that the size of the filtered dataset is much smaller than that of the original dataset; the mining time can thus be significantly reduced[1].

5. The proposed algorithm:

Lattice structure

Lattice structure is used to store all frequent itemsets in the dataset.



A lattice structure is constructed by a set of nodes in which each node contains an itemset and other information

Each node at the first level of the lattice is a tuple in the form:

{id,itemset,(obidset1,……,obidsetk),(o1,…..,ok),pos,total,traverse,childrenEC,children)

Each node at the second level of the lattice is a tuple, as follows:

{id,itemset,(diffset1,……,diffsetk),(o1,….., ok),pos,total,traverse,childrenEC,children)

6. Parallel Computing:

Single systems with many processors work on same problem.

- To be run on a single computer having a single Central Processing Unit (CPU);
- A problem is broken into a discrete series of instructions.
- Instructions are executed one after another. Only one instruction may execute at any moment in time

The major algorithms used in parallel computing are:

1. count distribution
2. Data distribution
3. 3.Candidate distribution
4. 4.Eclat

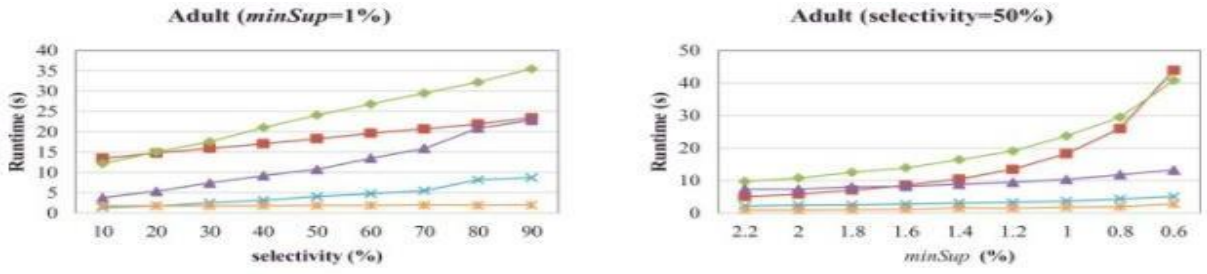
Why Parallel computing:

Parallel computing is complex on any aspect. The reason for parallel computing is save time, solve larger problems and provide concurrency.

Table 1:Characteristics of datasets:

Dataset	# attributes	# classes	Class distribution (%)	# distinctive values	# objects
Adult	13	2	(75, 25)	63	30,162
Breast	12	2	(66, 34)	737	699
Chess	37	2	(75, 25)	76	3196
Connect-4	43	3	(99.86, 0.11, 0.03)	130	67,557
German	21	2	(96, 4)	1077	1000
Hypothyroid	26	2	(5, 95)	1204	3162
Ionosphere	35	2	(8, 39, 46, 7)	209	351
Lymph	19	4	(1, 55, 41, 3)	63	148
Mushroom	23	2	(52, 48)	119	8124
Nursery	9	5	(33.33, 32.92, 0.01, 31.21, 2.53)	32	12,959
Pumsb	74	5	(94.85, 0.09, 1.38, 3.27, 0.40)	2113	49,046
Tic-tac-toe	10	2	(35, 65)	29	958
Vehicle	19	4	(25, 26, 26, 23)	1434	846
Vote	17	2	(62, 38)	50	434

Execution time for adult dataset:



(a) Runtime vs selectivity (b) Runtime vs *minSup*

7. Experiments:

All experiments were conducted on a computer with an Intel Core i7-5500U CPU at 3.00 GHz and 8 GB of RAM running OS Windows 8.1 (64-bit)[1]. The experimental datasets were obtained from the UCI Machine Learning Repository (http://mlearn.ics.uci.edu).

Experimental datasets:

Table1 shows the main characteristics of experimental datasets. It shows the number of attributes (including the class attribute), the number of class labels, the class distribution, the number of distinctive values (i.e. the total number of distinct values in all attributes), and the number of objects [1].

Memory usage –Comparison:

The memory usage is determined by the total amount of memory which stores all nodes in the tree for CCAR and in the lattice for LD-CARM-IC. The memory reduction of LD-CARM-IC over CCAR is calculated by the below equation[1].

$$\text{Memory reduction} = 100\% - \text{MemoryofLD-CARM-IC} / \text{MemoryofCCAR} \times 100\%$$

Mining time

Experiments were conducted to determine the efficiency of the algorithm.Miming time is compared using the support for itemsets based on the intersections among transaction identifiers . To initialize itemset constraint β, we define the selectivity of a constraint as the ratio of the number of items selected to be the constraint against the total number of items. A constraint with 0% selectivity means no items, while a constraint with 100% selectivity is the one selecting all the items (distinctive values) in the dataset[1].

8. Conclusions:

This study has introduced an efficient method for mining CARs with the itemset constraint. The lattice structure is used to store all frequent itemsets in the dataset while four theorems are used to quickly prune infrequent itemsets, update the paternity relations among nodes, and calculate the *diffset* of a node. The *diffset* concept is also used to reduce the memory consumption. parallel programming is to execute code efficiently, since parallel programming saves time, allowing the execution of applications in a shorter wall-clock time. Apply parallel computing to the problem of mining CARs with the itemset constraint to speed up the process on very large datasets.

REFERENCES:

[1]. Dang Nguyen, Loan.T.T. Nguyen,Bay Vo “Efficient mining of class association rule with itemset constraint”

[2]. Loan T.T.NguyenacNgoc ThanhNguyenbc “An improved algorithm for mining class”

[3]. Liu B., Hsu W., Ma Y”Integrating classification and association rule mining

[4]. Li W., Han J., Pei J.CMAR: accurate and efficient classification based on multiple class-association rules The IEEE International Conference on Data Mining (ICDM 2001), IEEE (2001),pp. 369-376

[5]. N. Abdelhamid, A. Ayesh, F. Thabtah, S. Ahmadi, W. HadiMAC: a multiclass associative classification algorithm J. Inf. Knowl. Manage., 11 (2012), pp. 1-10