MAKER: A New Algorithm in Finding Frequent Itemsets

Dr. Masoud Yaghini[1], **Kaveh Rasouli Chizari**[2], **Mahsa Mortazavi**[2], **Erfan Khaji**[3], **and Mahyar Hoseynzadeh**[3]

[1] School of Railway Eng., Iran University of Science & Technology, Tehran, Iran
[2] School of IT & Electronics, Iran University of Science & Technology, Tehran, Iran
[3] School of Railway Eng., Iran University of Science & Technology, Tehran, Iran

Abstract *- Many frequent itemset algorithms have been proposed within recent years most of which have some difficulties when minimum support is low or the dataset is dense. In this paper, we present a novel algorithm named Maker using data storing based on prime numbers and Dynamic Blocking which addresses the blocks in which a product is purchased with at least α probability. We show that dynamic blocking drastically cut down the size of memory required to store dense datasets. We demonstrate how maker incorporated into previous mining methods increase the performance significantly.*[1]

Keywords: Data mining, Association rules, Dynamic-Blocking, Maker algorithm

1 Introduction

Frequent Itemsets Mining (FIM) is a classical issue in data mining which has been the subject of many investigations within recent years. However, there is still one unsolved problem which is FIM in dense datasets with low minimum support. Historically, investigating on three major areas has led researchers to novel approaches to optimize former algorithms: Code optimization, candidate generation and data structure. Most reputed Apriori [1] has presented a practical candidate generation approach used by many later procedures, Patricia, kdci and lcm [2, 3, 4] to name a few. The idea of changing data structure was first discussed in Fp-growth [5] and developed in Eclat [6] and their descendants [7], [9]. Beside vertical and horizontal datasets, Prime Number dataset [10] is also considered as useful mean to decrease the time consumed in referring the main dataset. As Apriori and the other algorithms which investigated candidate generation need refer to the main memory for many times, they usually demands the other approaches such as data structure to achieve a satisfactory performance. Moreover, decreasing the value of minimum support will result in enlarging the volume of all of the data structures which would make trouble in generating and searching both the structure and candidates. Hence, the so-called problem cannot be handled effectively by the previous algorithms. In this paper, we present an innovative approach named Maker using two novel procedures to solve the problem, *Dynamic Blocking* and *Prime number* dataset which led us to a perfect result in mining dense datasets with low minimum support.

1.1 Problem Statement

Let us fix notations for the frequent itemset mining problem in the rest of this section. Let A be a set, called *set of items* or *alphabet*. Any subset $X \in \rho(A)$ of A is called an *itemset*. Let $\Gamma \subseteq \rho(A)$ be a multiset of itemsets, called *transaction database*, and its elements $T \in \Gamma$ called *transactions*. For a given itemset $X \in \rho(A)$, the set of transactions that contain X

$$\Gamma(X) := \{ T \in \Gamma | X \subseteq T \} \quad (1)$$

is called *(transaction) cover of X in* Γ and its cardinality

$$\mathrm{Sup}_{\Gamma}(X) := |\Gamma(X)| \quad (2)$$

(absolute) support of X *in* Γ. An *(all) frequent itemset mining task* is specified by a dataset Γ and a lower bound minsup ⊆ N on support, called *minimum support*, and asks for enumerating all itemsets with support at least min- sup, called *frequent or (frequent) patterns*.

2 Dynamic Blocking

Lets TID_i and TID_j be two transactions in a dense vertical dataset where:

$$j > i . \quad (3)$$

We name the pair of transactions $\{i, j\}$ a *Block* of product *A*, and we show this block in the form of $\{i,j\}_A^k, k = 1,2,..,n$ where *k* is the number of blocks. Consider the situation where at least α percent of a block contains the product *A*. Therefore, we can represent a vertical dataset in the following series:

$$A = \{TID_i, TID_j - TID_k, TID_l - TID_m, TID_n - TID_o, TID_p - TID_q, TID_r\}. \quad (4)$$

In the problem of FIM with the minimum support of *min-sup*, finding the frequency of an n-itemset in a given dataset is desired. Using *dynamic blocking* approach, the problem is reviewed in the following steps:
Let's

$$\bigcap_{m \subset X}^{o \subset I} \{i,j\}_m^o = \{a,b\} . \quad (5)$$