Report on Weighted Frequent Itemset Mining

Ritesh Yadav

June 2020

1 Introduction

Frequent item Mining(FIM) is the mining of frequently occurring ordered events or subsequences as pattern in sequence database. In recent decades, extensions of FIM such as **weighted frequent itemset mining (WFIM)** and **frequent itemset mining in uncertain databases (UFIM)** have been proposed. WFIM considers that items may have distinct weight/importance. It can thus discover itemsets that are more useful and meaningful by ignoring irrelevant itemsets with lower weights.
Even though, WFIM consider different weights of each item during the mining process, it is not enough to reflect the real world enviroment where the weight of an item can vary with time. In our real world scenarios, the significance (weight) of an item might be widely affected by many factors. Customer's buying behaviors (or interests) are changing with time, so they affect the significances(weights) of products in retail markets.
We will further look into some major researches in the area of WFIM.

2 Research Works

Many research papers and findings related to them for weighted frequent itemsets generation provides different ways of assigning weights to itemsets in transactions. Some of the popular refrences are mentioned below :

1. **Efficient Mining of Weighted Quantitative Association Rules and Characterization of Frequent Itemsets** by Arumugam G and Vijayakumar V.K
2. **A Weighted Frequent Itemset Mining Algorithm for Intelligent Decision in Smart Systems** by Xuejian Zhao, Xinhui Zhang, Pan Wang, Songle Chen and Zhixin Sun
3. **Mining Weighted Frequent Itemsets without Candidate Generation in Uncertain Databases** by Chun-Wei Jerry Lin, Wensheng Gan, Philippe Fournier Viger and Tzung-Pei Hong
4. **An Efficient Algorithm for Mining Weighted Frequent Itemsets Using Adaptive Weights** by Hung Long Nguyen
5. **Weighted Frequent Itemset Mining with a weight range and a minimum weight** by Unil Yun and John J. Leggett

3 Analysis

Research's in **[1,2,3]**, suggests assigning distinct weights for each item in the whole dataset and continue with it in the whole dataset of transactions. We can't assign different weights to items in different transactions, the weight assigned must be initialize by the user earlier. They have also consider **uncertain databases** i.e. each item is assigned some uncertainty probability of getting picked in that particular transaction.

Example: Let the total number of items available to user are A, B, C, D.

Item	Weight
A	0.5
B	0.36
C	0.8
D	0.2

Table 1: Weight Table

Transactions by user :

TID	Transaction(item,probability)
T1	(A, 0.42), (C, 0.31), (D,0.12)
T2	(B, 0.23), (D, 0.56)
T3	(A, 0.48), (B, 0.2), (C, 0.5), (D, 0.35)
T4	(A, 0.82), (B, 0.3)

Table 2: Transaction Table

Here probability represents probability of getting picked.

In **[4]**, they have proceeded correspondingly as above by removing uncertainty probability and adding the option of assigning weights to items batch-wise known as **Adaptive weights**. They have divided the transaction database into **k-batches** where each batch have distinct-weight items.
Example: Let the total number of items available to user are A, B, C, D.

- **Batch-1** :

Item	Weight
A	0.3
B	0.7
C	0.5
D	0.2

Table 3: Weight Table

Transactions by user :

TID	Transaction(item)
T1	B C D
T2	A B D

Table 4: Transaction Table

- **Batch-2** :

Item	Weight
A	0.6
B	0.1
C	0.6
D	0.3

Table 5: Weight Table

Transaction by User:

TID	Transaction(item)
T3	A B
T4	B A C
T5	A D B

Table 6: Transaction Table

Transactions in above example take place in two different batches.

In [**5**], the author has taken a **Weight Range(WR)** attribute, and weights of the items must lie within this WR in transactions. **Their main approach is to push weight constraints into the pattern growth algorithm and provide ways to keep the downward closure property.**

4 Conclusion

The findings of the research [**4**] somewhat match our area of interest as we can proceed by keeping the number of transactions in each batch equal to one for assigning distinct weights to items in each transaction.