CSC 440-Data Mining Project-1

Implementation of Apriori and FP-Growth Algorithms

Sai Pranav Chittella

University of Rochester
(NetID:schittel)

ABSTRACT

In Data Mining, Association Rule Mining is a standard and well researched technique for locating interesting relations between variables in large databases. Association rule is used as a precursor to different Data Mining techniques like classification, clustering and prediction. The aim of the project is to gauge the performance of the Apriori algorithm and Frequent Pattern (FP) growth algorithm by implementing them on the adult data set and comparing their capabilities with various support.

KEYWORDS

Apriori, FP-Growth, Adult data set, Data Mining, Associative Rule Mining

1 INTRODUCTION

Data Mining is a promising and flourishing frontier in analysis of data and also the results of analysis has several applications. Data Mining, also popularly referred as Knowledge Discovery from Data (KDD), is a convenient extraction of patterns representing knowledge implicitly keep or captured in huge databases, data warehouses, the Web, data repositories, and information streams.

Association Mining aims to extract correlations, frequent patterns, and association structures among set of items or objects in transaction data based relational databases or different data repositories. Two statistical measures that are important in the Association Rule Mining are Support and Confidence. Support should be measured as to how often it should occur in the database. Confidence may well be gauged to seek out the strength of the rule. The Association rules are interesting if they satisfy each a minimum Support threshold and a minimum Confidence threshold

Association rules describe how often items are purchased together. For example, an association rules “beer, chips (80%)” states that four out of five customers that bought beer also bought chips. Such rules can be useful for decisions concerning product pricing, promotions, store layout and many others. [1]

2 ASSOCIATION RULE MINING

An Association rule mining[3] is an expression of the form X-->Y means that whenever X seems, Y also tends to appear. X and Y are itemsets. The itemsets are nothing but a collection of database items. X is stated as the rule’s antecedent and Y as the consequent of the rule.

We have two measures to calculate the strength of the association. Support is the proportion of transactions in an exceedingly information that satisfy the rule.Confidence denotes the chance of Y being a true subject to X or P (Y|X).

Generally association rule mining contains following steps:

- The set of candidate k-itemsets is generated by 1-extensions of the large (k -1) itemsets generated in the previous iteration.
- Supports for the candidate k-itemsets are generated by a pass over the database.
- Itemsets that do not have the minimum support are discarded and the remaining itemsets are called large k-itemsets.

The algorithms Apriori and the FP-Growth are applied on the UCI Adult [2]dataset.

Section 2.1 talks about Apriori and 2.2 talks about FP-Growth algorithm.

2.1 Apriori Algorithm

Agrawal and Srikant (1994) firstly proposed Apriori algorithm[4]. This algorithm is based on Apriori property which states “every sub (k-1)-Itemset of frequent k-Itemset must be frequent”

Apriori could also be a candidate generation algorithmic program and issue in an extremely level wise fashion.It uses breadth first search and a tree structure to count candidate itemsets efficiently.

The Apriori property is broadly divided into two steps:

1. *Join step*: - Ck is generated by combining lk–1 with itself.
2. *Prune Step*: - Any (k – 1) item set that’s not frequent cannot be a set of a frequent k item set.

Pseudo code for Apriori Algorithm:

Apriori(T,min_Support)//T is the database and min_Support is the minimum support
L1 = {frequent items}
For (k= 2; Lk-1 ! =∅; k++){
Ck= candidates generated from Lk-1
For each transaction t in database **do{**
Increment the count of all candidates in Ck that are contained in t
Lk = candidates in Ck with min_Support
}
}
Return **UkLk**
}

2.2 FP-Growth Algorithm

To break the two drawbacks of Apriori algorithm, FP-growth algorithm[5] is used. FP-growth requires constructing FP-tree. For that, it requires two passes. FPgrowth uses divide and conquer strategy. It requires two scans on the database. It first computes a list of frequent items sorted by frequency in descending order (F-List) and during its first database scan. In the second scan, the database is compressed into a FP-tree.

The frequent itemsets are generated with only two passes over the database and without any candidate generation process. There are two sub processes of frequent patterns generation process which includes: construction of the FP-tree and generation of the frequent patterns from the FP-tree.FP-tree is constructed over the data-set using 2 passes.

The steps are discussed below,

- Step1:
  Pass-1: Scan the information and realize support for every item and discard rare things. Then type frequent things in downward order that is based on their support.By exploitation this order we will build FP-tree, so common Prefixes will be shared.
- Step 2:
  Pass-2: Here nodes correspond to things and it is a counter.FP-growth reads one dealings at a time then maps it to a path. Mounted order is employed, so methods will overlap once transactions share the things.
- Step 3: In this final step, counters are incremented. Some pointers are maintained between nodes that contain identical item, by creating on an individual basis coupled lists. The lot of methods that overlap, higher the compression. FP-tree could slot in memory. Finally, frequent itemsets are extracted from the FP-tree.

.

3 GENERAL ANALYSIS OF ALGORITHMS

3.1 Apriori Algorithm

The following table presents the performance survey of the Apriori algorithm,

S.No	Performance Factor	The Apriori way
1	Data Structure	Array
2	Memory Utilization	Memory requirement is huge
3	No. Of Scans required	Multiple scans are required to generate the candidate set.
4	Execution time	More time due to wastage of time in generating candidates at every step.
5	Technique	Use Apriori property in association with the join and prune method

3.2 FP-Growth Algorithm

The following table presents the performance survey of the FP-Growth algorithm,

S.No	Performance Factor	The FP-Growth way
1	Data Structure	Tree
2	Memory Utilization	Due to no candidate generation at every step, less memory is required
3	No. Of Scans Required	The database is scanned twice
4	Execution time	Lesser compared to apriori(presented in the results)
5	Technique	It constructs conditional frequent pattern tree and conditional pattern base from database which satisfy the minimum support

4 RESULTS

Experimental analysis is performed on Adult data set, The data set contains 48842 itemsets and 45223 number of transactions.

The minimum support is given as 23% or 0.23, and the calculations and the respective output frequent itemsets are indicated in the OUT_FILE.pdf .

Alongside, for the comparison between the Apriori and FP-Growth, various support values are substituted, and their performance is observed.

The runtime synthesis(in seconds) is indicated in the table below,

Minimum Support	Apriori	FP-Growth
23%	154.2	1.503
50%	164	1.717
60%	153	1.372

To extend my understanding, I have tried few other data sets which are popular for Frequent itemset mining, and the particular one is Mushroom data set[].

CONCLUSION

The algorithms were implemented using java, and a special data structure FP-Tree[5] for FP-Growth algorithm.

Although, a learning project, it is evident that classifiers work better with the adult data set and the same is indicated in their description on the website. As an extension to the Apriori, I have written the code all over again, and have tested it. I cannot judge whether it works accurately or not, but have tried attempting it. Alongside, I have studied about other data sets and algorithms, which are to be implemented in the future such as ECLAT[6],RELIM[7],etc.

REFERENCES

[1] Han, J., Kamber, M.,"Data Mining concepts and techniques", Elsevier Inc., Second Edition, San Francisco, 2006.

[2] https://archive.ics.uci.edu/ml/datasets/adult . Adult Data Set:UCI Machine Learning Repository

[3] Sotiris Kotsiantis and Dimitris Kanellopoulos,"Association Rules Mining: A Recent Overview", GESTS International Transactions on Computer Science and Engineering, Vol.32, No: 1, pp. 71-82, 2006

[4] https://en.wikipedia.org/wiki/Apriori_algorithm. Apriori Algorithm.

[5] http://www.cis.hut.fi/Opinnot/T-61.6020/2008/fptree.pdf : FP-Tree.

[6] https://www.slideshare.net/deepa15/eclat-37310304/ :ECLAT Algorithm.

[7] http://www.borgelt.net/doc/relim/relim.html : RELIM Algorithm