Implementation of Sales Analysis System for Fancy Shop based on Association Rule Mining

Khine Zin Oo, Aye Aye Khaing
Computer University, Kyaing Tong
khinezinoo8@gmail.com,khaingaa@gmail.com

Abstract

One of the important problems in data mining is discovering association rules from databases of transactions where each transaction consists of a set of items. The most time consuming operation in this discovery process is the computation of the frequency of the occurrences of interesting subset of items (called candidates) in the database of transactions. The proposed system presents the FP-Growth algorithm to avoid or reduce candidate generation. FP-growth method is efficient and scalable for mining both long and short frequent patterns without candidate generation. It not only heirs all the advantages in the FP-growth method, but also avoid its bottleneck in database size dependence when constructing the frequent pattern tree (FP-tree).It greatly reduces the need to traverse the database. This process analyzes customer buying habits by finding associations between the different items that customers place in their shopping baskets. The proposed system tends to analysis on the Sales System of Fancy Shop using FP-growth algorithm under association rules mining. The system evaluates more suitable products to sell and how to display them according to their purchasing products.

Keywords: Data Mining; Association Rule; Apriori Algorithm; FP-growth Algorithm; Improved Association Rule;

1. Introduction

In recent years the sizes of databases has increased rapidly. This has lead to a growing interest in the development of tools capable in the automatic extraction of knowledge from data. The term Data Mining or Knowledge Discovery in Database has been adopted for a field of research dealing with the automatic discovery of implicit information or knowledge within databases. The implicit information within databases, and mainly the interesting association relationships among sets of objects, that lead to association rules, may disclose useful patterns for decision support, financial forecast, marketing policies, even medical diagnosis and many other applications.

Association rule mining finds interesting association or correlation relationships among a large dataset of data items. A typical example of association rule mining is the market basket analysis. This process analyzes customer buying habits by finding associations between the different items that customers place in their shopping baskets. This thesis tends to analysis on the Sales System of Fancy House using FP-growth algorithm under association rules mining. Mining frequent patterns in transaction databases, time-series databases, and many other kinds of databases has been studied popularly in data mining research. Most of the previous studies adopt an Apriori-like candidate set generation-and-test approach. However, candidate set generation is still costly, especially when there exist prolific patterns and/or long patterns.

To break the two bottlenecks of Apriori series algorithms, some works of association rule mining using tree structure have been designed. FP-Tree, frequent pattern mining, is another miles tone in the development of association rule mining, which breaks the two bottlenecks of the Apriori.

The frequent itemsets are generated with only two passes over the database and without any candidate generation process. FP-Tree was introduced by Han et al in [3]. By avoiding the candidate generation process and less passes over the database, FP-Tree is an order of magnitude faster than the Apriori algorithm. The frequent patterns generation process includes two sub processes: constructing the FP-Tree, and generating frequent patterns from the FP tree.

This paper is organized as follows: The related work of Association Rule mining is described in Section 2. Section 3 presents the Association Rule Mining. Section 4 presents the FP-growth algorithm. Proposed system design has been introduced in

Section 5 and System implementation is described in Section 6. This paper is concluded in Section 7.

- To know what are the Association Rules Mining
- To understand the detail meaning of FP-growth Algorithm and how it's work
- To understand how to apply the FP-growth Algorithm in the Sales Analyzing Systems
- To see how the system is improved by applying those algorithm in the development of the system
- To estimate which products are better demand and associate to sell
- To improve the profit of Fancy Shop

2. Related Work

Frequent-pattern mining plays an essential role in mining associations. Most of the previous studies, adopt an Apriori-like approach, which is based on the anti-monotone Apriori heuristic [1]: if any length k pattern is not frequent in the database, its length (k + 1) super-pattern can never be frequent. The essential idea is to iteratively generate the set of candidate patterns of length (k+1) from the set of frequent-patterns of length k (for $k \geq 1$), and check their corresponding occurrence frequencies in the database.

The Apriori heuristic achieves good performance gained by (possibly significantly) reducing the size of candidate sets. However, in situations with a large number of frequent patterns, long patterns, or quite low minimum support thresholds, an Apriori-like algorithm may suffer from the following two nontrivial costs:

- It is costly to handle a huge number of candidate sets.
- It is tedious to repeatedly scan the database and check a large set of candidates by pattern matching, which is especially true for mining long patterns.

3. Association Rule Mining

Association rules are used to identify relationships among a set of items in database. These relationships are not based on inherent properties of the data themselves (as with functional dependencies), but rather based on co-occurrence of the data items. Association rule mining has a wide range of applicability such Market basket analysis, Medical diagnosis/ research, Website navigation analysis, Homeland security and so on. [2]

Mining association rules are to find interesting association or correlation relationships among a large set of data, i.e., to identify sets of attributes values (predicate or item) that frequently occur together, and then formulate rules that characterize these relationships. Mining association rules is composed of the following two steps –

- Discover the large itemsets, i.e., all sets of itemsets that have transaction support above a predetermined minimum support s.
- Use the large itemsets to generate the association rules for the database.

The overall performance of mining association rules is in fact determined by the first step. Let I= {il, i2, . . ., im} be a set of literals, called items. Let D be a set of transactions, where each transaction, T is a set of items such that $T \subseteq I$. Each transaction is associated with an identifier, called TID. Let X be a set of items. A transaction T is said to contain X if and only if $X \subseteq T$. An association rule is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$ and $X \cap Y = \phi$. The rule $X \Rightarrow Y$ holds in the transaction set D with confidence c if c% of transactions in D that contain X also contain Y. The rule $X \Rightarrow Y$ has support s in the transaction set D if s % of transactions in D contains $X \cup Y$.

4. FP-Growth Algorithm

Information from transaction databases is essential for mining frequent patterns. [4] Therefore, if we can extract the concise information for frequent pattern mining and store it into a compact structure, then it may facilitate frequent pattern mining. FP-tree stores complete but no redundant information for frequent pattern mining. Steps of constructing FP-Tree are shown as follows:

Since only the frequent items will play a role in the frequent-pattern mining, it is necessary to perform one scan of transaction database TDB to identify the set of frequent items (with frequency count obtained as a by-product).

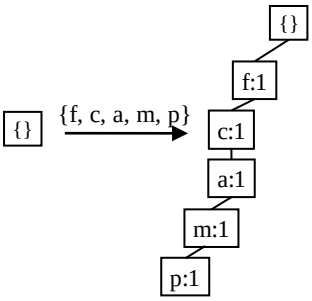
- If the set of frequent items of each transaction can be stored in some compact structure, it may be possible to avoid repeatedly scanning the original transaction database.
- If multiple transactions share a set of frequent items, it may be possible to merge the shared sets with the number of occurrences registered as count. It is easy to check whether two sets are identical if the frequent items in all of the transactions are listed according to a fixed order.
- If two transactions share a common prefix, according to some sorted order of frequent items, the shared parts can be merged using one prefix structure as long as the count is registered properly. If the frequent items are

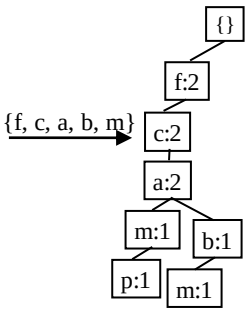
sorted in their frequency descending order, there are better chances that more prefix strings can be shared.

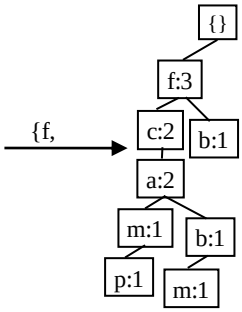
Table 1: Sample Transaction database

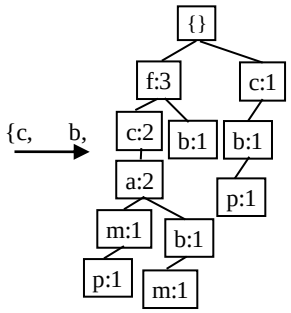
TID	Items Bought	(Ordered) Frequent Items
100	f, a, c, d, g, i, m, p	f, c, a, m, p
200	a, b, c, f, l, m, o	f, c, a, b, m
300	b, f, h, j, o	f, b
400	b, c, k, s, p	c, b, p
500	a, f, c, e, l, p, m, n	f, c, a, m, p

Construct FP-tree.









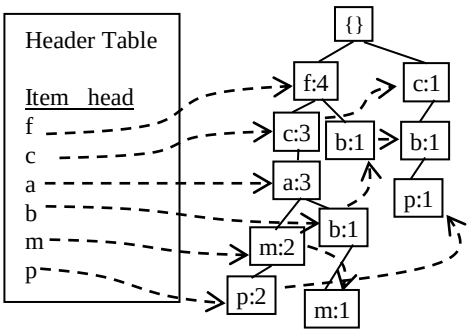


Figure 1: FP Tree Construction of Table 1

5. Proposed System

This system presents the FP-Growth algorithm, improvement of Apriori algorithm of association rule.

FP Tree algorithm solves the bottle-neck of Apriori algorithm. This proposed system is to analyze the sale of Fancy Shop by using FP-growth algorithm. There are 30 fancy item types available at the Fancy shop. Those items information will be stored at the Item Database. When the customer comes and buys the items, the cashier will enter the sales information from the Sales Entry Form and the system will stored those sales data at the sales table. When the manager wants to view which item sets are most popular and sold out most frequently (hot item sets), the manager only needs to choose the sales date(s).

The system will automatically find the most paired item sets by using FP-growth algorithm and show the most frequent paired sold out items to the manager. By viewing the outcome results (sales analyzing report), the manger can make future market plans for his/her shop. Figure 2 presents the proposed system design.

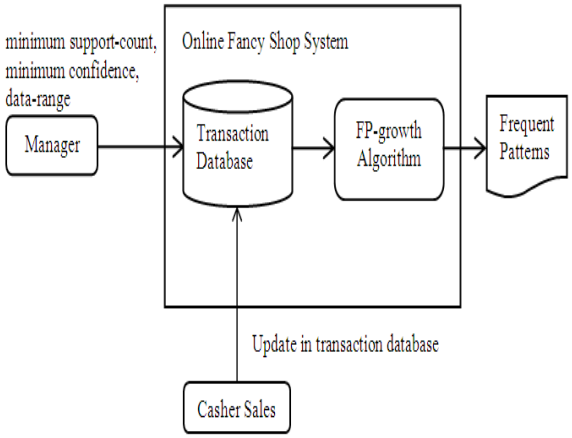


Figure 2: Proposed System Design

6. System Implementation

This system is implemented as web-based sales system. It is developed using Microsoft Visual Studio .Net 2008. ASP .Net C# is used as the programming language. Microsoft Access 2003 is used to store the transaction data. This system consists of two modules, Manager module and Cashier module. Manager is responsible for adding, deleting and updating items and other system administration process. Sales data analysis and querying transaction data can also be done by Manager only. Cashier is for entering sales data into the system. Detailed process flow of the system is shown in Figure 3.

The database design of this system includes item information and transaction data. Item information contains Brand and Item Category. In this system, patterns generated based on item category, for example customers who bought rings also bought earrings. The database design of the system is shown in Figure 4.

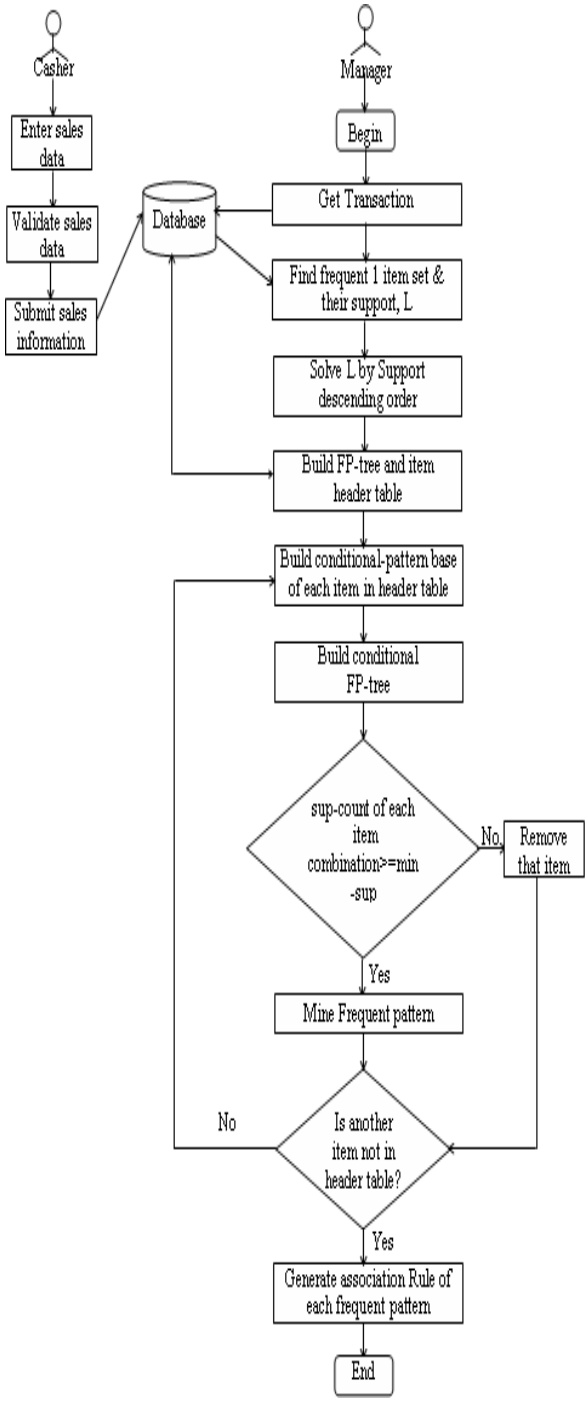


Figure 3: Process Flow of the System

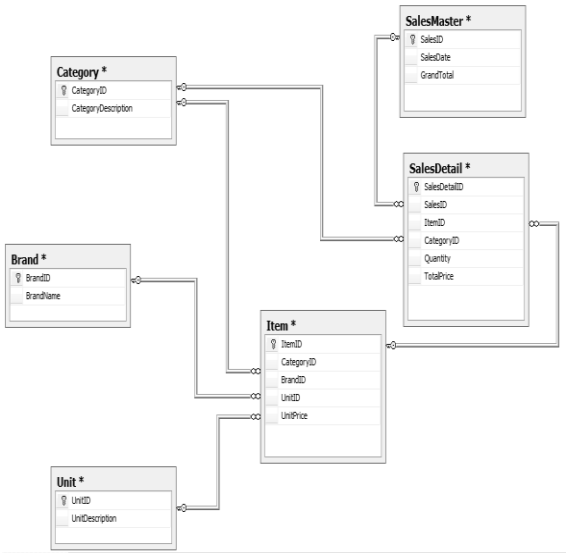


Figure 4: Database Design

6.1. FP-Growth Algorithm Implementation

FP-Growth algorithm is implemented by building FP-Tree. Frequent pattern tree is constructed as follows:

First, the first scan of the database derives a list of frequent items, in which items are ordered in frequency-descending order. This ordering is important since each path of a tree will follow this order.

Second, the root of a tree is created and labeled with "null". The FP-tree is constructed as follows by scanning the transaction database DB the second time.

FP-Tree Construction Algorithm

Input: A transaction database DB; minimum support threshold, min_sup.

Output: The complete set of frequent patterns.

Method: Call FP_growth (FP_tree, null)

Procedure FP_growth (Tree, a)
{
(1) If tree contains a single path P
(2) then { Let P be the single path of tree
(3) Let Q be the multi path with the top branching node replaced by a null root;
(4) For each combination (denoted as b) of the nodes in the path P do
(5) Generate pattern b U a with support = minimum support of nodes in b.
(6) Let the frequent-pattern-set (P) be the set of patterns so generated ;}
(7) Else let Q be Tree;
(8) For each item a_i in Q do {
(9) Generate pattern b = a_i U a with support= a_i.sup;
(10) Construct b's conditional pattern-base and then b's conditional FP-tree Tree b;
(11) If Tree b ≠ 0
(12) then call FP-growth (Tree b, b);
(13) Let frequent-pattern-set (Q) be the set of patterns so generated;
}

6.2. Experimental Result

This system is tested with 1000 transactions, with longest pattern in 10 item-lengths. Experimental results show that FP-Tree structure outperforms all existing available algorithms in all common data mining problems. We have tested the transaction data in different minimum support and minimum confidences. The processing times of the FP-Growth algorithm and Apriori Algorithm are shown in Table 2. Experimental result shows FP-growth algorithm runs exponentially faster than Apriori algorithm.

Table 2: Processing times of FP-Growth Algorithm and Apriori Algorithm (Time in milli-seconds)

No.	Min-sup	Min-conf	FP-Growth	Apriori
1	3	0.75	324	2389
2	4	0.75	245	1874
3	3	0.85	310	1980
4	4	0.85	234	1789

Different minimum support and minimum confidence produces different accuracy values. The accuracy of this system is shown in Table 3:

Table 3: Accuracy for Different minimum support and different confidence

No.	Min-sup	Min-conf	Accuracy (%)
1	3	0.75	0.85
2	4	0.75	0.87
3	3	0.85	0.84
4	4	0.85	0.88

7. Conclusion

This paper tends to analyze on the Sales System of Fancy Shop using FP-growth algorithm under association rules mining. By using FP-growth algorithm, it is only need to scan database only twice and it does not generate candidate sets. The system evaluates more suitable products to sell and how to display them according to their purchasing products. By using this system, business manager can make decisions to increase the sales and profits.

8. References

[1] Agarwal, R., Aggarwal, C. and Prasad, V. V. V., "Depth-First generation of large itemsets for association rules". IBM Tech. Report RC21538, July 1999.

[2] Agrawal, R. and Srikant, R., "Fast algorithms for mining association rules. In VLDB'94, pp. 487-499.

[3] Han, J. W. and Pei, J., "Mining Frequent Patterns without Candidate Generation: A Frequent-Pattern Tree Approach", Data Mining and Knowledge Discovery, 8, 53–87, 2004, 2004 Kluwer Academic Publishers. Manufactured in The Netherlands.

[4] Margahny, M.H. and Mitwaly, A.A. "Fast Algorithm for Mining Association Rules", Proceeding in AIML 05 Conference, 19-21 December 2005, CICC, Cairo, Egypt.

[5] Park, J.S., Chen, M.S. and Yu, P.S. "An effective hash-based algorithm for mining association rules", In SIGMOD'95, pp. 175-186.