A Data Mining Framework for Extracting Product Sales Patterns in Retail Store Transactions Using Association Rules: A Case Study

Mirzaei.Afshin [1], Sheikh.Reza[2]

[1]Department of Industrial Engineering and Management, Shahrood University of Technology ,SHAHROOD, IRAN,
Mirzaei.Afshin@gmail.com
[2]Faculty of Industrial Engineering and Management Shahrood University of Technology, SHAHROOD, IRAN,
resheikh@Shahroodut.ac.ir

Abstract: Widespread use of bar codes for most commercial products, the computerization of many businesses, and the advance in data collection tools have provided us with huge amount of retail data and data sets. This may be potentially valuable but currently untapped. These large datasets need to be analyzed for useful information, In a changing competitive environment, the usage of data mining to tap the potential knowledge and its decision-supporting applications would benefit organizations, businesses and individuals by supporting decision making and providing valuable knowledge. Association rules is a recent data mining technique to discover affinities, in large transaction databases, between items frequently purchased together. It has been claimed that the discovery of frequent sets of items is well suited for applications of market basket analysis to discover regularities in the purchase behavior of customers. This paper elaborates upon the use of association rule mining in extracting patterns that occur frequently within a dataset and showcases the implementation of the FP-Growth algorithm in mining association rules from a real dataset obtained from a supermarket chain containing sales transactions of a retail store.

[Afshin Mirzaei,Dr.Reza Sheikh. **A Data Mining Framework for Extracting Product Sales Patterns in Retail Store Transactions Using Association Rules: A Case Study.** Journal of American Science 2012;8(9):304-308]. (ISSN: 1545-1003)]. http://www.jofamericanscience.org. 43

Keywords: data mining, association rules, FP-Growth algorithm, RapidMiner

1. Introduction

In the past, retailers saw their job as one of buying products and putting them out for sale to the public. If the products were sold, more were ordered. If they did not sell, they were disposed of.

(Blischok 1995) describes retailing in this model as a product oriented business, where talented merchants could tell by the look and feel of an item whether or not it was a winner. In order to be successful, retailing today can no longer be just a product-oriented business. According to (Blischok 1995), it must be a customer-oriented business and superior customer service comes from superior knowledge of the customer. Since almost all mid to large size retailers today possess electronic sales transaction systems, retailers realize that competitive advantage will no longer be achieved by the mere use of these systems for purposes of inventory management or facilitating customer check-out. In contrast, competitive advantage will be gained by those retailers who are able to extract the knowledge hidden in the data, generated by those systems, and use it to optimize their marketing decision making. In the highly competitive retail industry, one of the keys to gaining an edge is an efficient shelf allocation system where shelf space is often the retailer's scarcest resource. As the number of brand lines continually increases, allocating products to the supermarket shelf in the best possible arrangement poses challenges to the retailer. Within the retail industry, user interest in shelf-space allocation is found to be very high. Retailers need frequently make decisions about which products to display (assortment) and how much shelf space to allocate these products (allocation) (Borin, N. and Farris 1995). Product assortment and shelf space allocation are two important issues in retailing which can affect the customers' purchasing decisions. Through the proficient shelf space management, retailers can improve return on inventory and consumer's satisfaction, and therefore increase sales and margin profit (Yang, M-H., and Chen 1999). A typical usage scenario for searching frequent patterns is the so called "market basket analysis" that involves analyzing the transactional data of a supermarket or retail store in order to determine which products are purchased together and how often and also examine customer purchase preferences. The Apriori algorithm introduced by (Agrawal et al) in 1994 is an efficient technique to generate all significant association rules between items in a database. Also Frequent pattern mining has become one of the most active topics in data mining for over a decade. It is a most time consuming process and plays an essential role in many data mining tasks that try to find interesting frequent patterns from databases, some examples are association rules mining, sequential pattern mining, structured pattern mining, and correlation mining, and so on. Since it was introduced in (1993 by Argawal), the frequent pattern mining has received a great deal of attention. Most of the previous efforts till 2000 adopted the Apriori-like candidate set generation and test approach, which is still very costly, especially with the existence of long patterns (J Han, J Pei ,Y Yin 2000).

In (2000, Jiawei Han) proposed an efficient FP-tree based mining method, FP-Growth, which adopts a divide and conquer way. The highly condensed data structure FPtree benefits FP-Growth with better performance than the Apriori-like algorithms. It is about an order of magnitude faster than Apriori algorithm. However, FP-Growth needs to recursively create huge amounts of conditional pattern bases and corresponding conditional FP-trees during mining process. When the dataset is huge, both the memory usage and computational cost are expensive, even the FP-tree based on the original database cannot meet the memory requirement.

Because of the huge storage but limited speed of the sequential FP-Growth, parallel algorithm becomes essential for large scale data warehouse mining. Most previous studies (Osmar R, Li Liu, Eric Li, Yimin Zhang) parallelized the FP-Growth algorithm in a shared memory system. Further performance can be expected from parallel execution on shared nothing environment, such as a computer cluster. As a very promising platform for high performance data mining, computer cluster has attracted a lot of attention recently. However, in the computer cluster, a parallel algorithm for complex data structure such as FP-tree is much harder to implement compared to the sequential program or shared memory parallel system. Section 2 of this paper describes the technique of association rule mining. Section 3 describes the working of the FP-Growth algorithm for generating significant association rules. Section 4 details our use of the RapidMiner data mining tool for generating association rules from a real dataset and our implementation of the FP-Growth algorithm to generate association rules from the real dataset.

2. Material and Methods

A. Association rules: An overview

Today, as a result of increased computing power and efficient algorithms, data mining techniques can be used to search efficiently for interesting information in large amounts of transaction data. A recent data mining technique for market basket analysis is *association rules* introduced by (Agrawal, Imielinski & Swami [1993]). The following is a formal introduction of this technique: Let $I = \{i1, i2, \ldots, ik\}$ 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 $T \subseteq I$. associated with each transaction is a unique identifier, called its *TID*. We say that a transaction T *contains* X, a set of some items in I, if $X \subseteq T$. An *association rule* is an implication of the form $X \Rightarrow Y$, where $X \subset I$, $Y \subset I$, and $X \cap Y = \varnothing$. 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 $X \cup Y$. Given a set of transactions D, the problem of mining association rules is to generate all association rules that have support and confidence greater than the user-specified minimum support (*minsup*) and minimum confidence (*minconf*).

Generating association rules involves looking for so-called *frequent sets* in the data. Indeed, the support of the rule $X \Rightarrow Y$ equals the frequency of the set $\{X,Y\}$. Thus by looking for frequent sets, we can determine the support of each rule (Mannila 1997).

Definition 1 Frequency of an itemset $s(X,D)$ represents the frequency of itemset X in D, i.e. the fraction of transactions of D that contain X.

Definition 2 Frequent itemset

An itemset X is called frequent in D, if $s(X, D) \geq \sigma$with σ the *minsup*. A typical approach [Agrawal, Mannila, Srikant, Toivonen & Verkamo 1996] to discover all frequent sets X is to use the knowledge that all subsets of a frequent set are also frequent. This insight simplifies the discovery of all frequent sets considerably, i.e. first find all frequent sets of size 1 by reading the data once and recording the number of times each item A occurs. Then form *candidate* sets of size 2 by taking all pairs $\{B, C\}$ of items such that $\{B\}$ and $\{C\}$ both are frequent.

The frequency of the candidate sets is again evaluated against the database. Once frequent sets of size 2 are known, candidate sets of size 3 can be formed; these are sets $\{B, C, D\}$ such that $\{B, C\}$, $\{B, D\}$ and $\{C, D\}$ are all frequent. This process is continued until no more candidate sets can be formed. Once all frequent sets are known, finding association rules is easy. Namely, for each frequent set X and each $Y \in X$ verify whether the rule $X \setminus \{Y\} \Rightarrow Y$ has sufficiently high confidence. The given algorithm has to read the database at most K+1 times, where K is the size of the largest frequent set. In the applications, K is small, typically at most 10, so the number of passes through the data is reasonable. For the problem being studied in this paper, K will be very small because the average customer buys only one or a few products from the automated convenience store during each shopping occasion.

Consequently, the technique of association rules produces a set of rules describing underlying purchase patterns in the data, like for instance *bread ⇒cheese* [support = 20% ; confidence = 75%]. Informally, support of an association rule indicates how frequent that rule occurs in the data. The higher the support of the rule the more prevalent the rule is. Confidence is a measure of the reliability of an association rule. The higher the confidence of the rule, the more confident we are that the rule really uncovers the true relationship in the data. It is clear that we are especially interested in association rules that have a high support and a high confidence. The FP-Growth (Frequent Pattern) algorithm for mining association rules, however, takes advantage of structure within the rules themselves to reduce the search problem to a more manageable size.

3. FP-GROWTH ALGORITHM

A. Problem Description

The following is a formal statement of frequent pattern mining (R. Agrawal and R. Srikant 1994): Let $I = \{i1, i2, ..., in\}$ be a set of literals, called items and n is considered the dimensionality of the problem. Let D be a set of transactions, where each transaction T is a set of items such that $T \subseteq I$. A transaction T is said to contain X, a set of items in I, if $X \subseteq T$. An itemset X is said to be frequent if its support (i.e. ratio of transactions in D that contain X) is greater than or equal to a given minimum support threshold. The frequent itemset X is called a frequent k-itemset if it contains k items. A 1-itemset is also called a frequent item.

B. FP-Growth Algorithm

FP-Growth uses a pattern fragment growth method to avoid the large number of scans on the database. It requires only two scans. The first is to accumulate the support of each item and then put the frequent items into a list, F-List, sorted in frequency descending order. In the second scan, the original database is compressed into a highly condensed FPtree, which stores the critical information of frequent patterns. After the construction of FP-tree, FP-Growth works in a divide and conquers way, decomposing the FP-tree into a group of independent conditional pattern bases of the frequent items. FP-Growth needs to build conditional FPtree according to each conditional pattern base. So far, the mining task is converted into a group of independent process, constructing and mining conditional FP-tree recursively. The pseudo code of mining on FP-tree is depicted in Figure 1.

```
procedure FP-Growth ($Tree$, $\alpha$)
{
if $Tree$ contains a single path $P$ then
for each $\beta$ = nodes combination in $P$ do
pattern = $\beta \cup \alpha$ ;
support = min(support of the nodes in $\beta$
);
else
for each $ai$ in the header of $Tree$ do
pattern $\beta = a\, i \cup \alpha$ ;
with support = $a\, i$. support ;
construct conditional pattern base of $\beta$
$TreeB$ = construct conditional FP-tree of
$\beta$
if $TreeB$!= Ø then
call FP-Growth( $TreeB$ , $\beta$ )
```

Figure 1. Pseudo code of FP-Growth algorithm

4. Empirical study

The empirical study is based on a data set of 32,215 sales transactions which came from time consuming data preprocessing using MATLAB software, acquired from a store over a period of 4 months in 2011. The store under study consists of over 2700 different items with a significant barcode for each item. With regard to the costs of each individual product in the assortment, detailed information on handling and inventory costs could not be obtained so these will be considered equal for all products and therefore these costs are not included in the model. Basically, the empirical study involves two important phases. In the first phase, structural purchase behavior under the form of frequent item sets is discovered by using the data mining technique of association rules. Then, in the second phase, the FP-Growth algorithm is used to select a hit list of products from the assortment. With data preprocessing using MATLAB software raw data eventuated in useful information by means of RapidMiner software which illustrated below.

Table 1 shows the sample form of raw data which include time of purchases and their relative barcodes:

TABLE 1. Time and Barcode

Time	Barcode
9:08:44	2.5E+11
16:22:48	2.7E+11
11:30:51	3.5E+11
18:17:54	7.7E+11
10:29:31	7.9E+11
9:47:13	2E+12
19:14:56	4E+12
-	-
-	-
-	-
-	-

Data preprocessing which done with MATLAB software include following steps: first it deleted the barcodes which is not so important like the items which has slow sale's circle or not enough profit and then it eventuate in matrix like below which is the format of input data for RapidMiner software:

Table 2 shows the sample form which is the output of MATLAB software. The quantity of purchased is not considered; only whether or not a particular item is purchased.

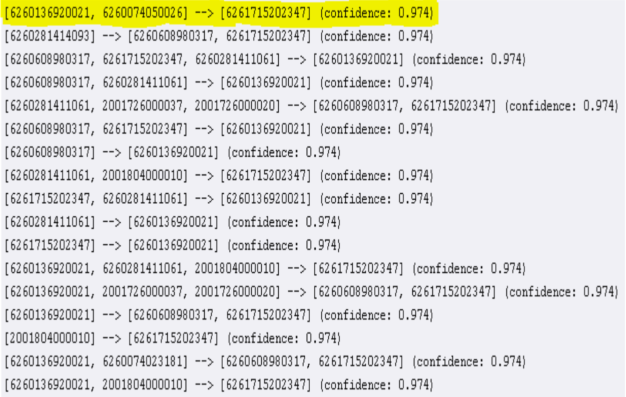
TABLE 2. Sample Transactional Data Form From Retail Store Data

2.50008E+11	7.61934E+11	2.00022E+12	6.00052E+12	9.00052E+12	.	4.00052E+12
0	1	0	0	1	.	1
1	1	0	1	0	.	0
1	0	0	0	1	.	1
1	1	0	1	1	.	0
1	0	1	1	0	.	1
0	1	1	0	1	.	1
.	.	.	.	.	.	.
.	.	.	.	.	.	.
.	.	.	.	.	.	.

Applying FP-Growth algorithm and then association rule generator on the real transactions and specifying the minimum confidence as 80%, the rules are obtained.

5. Results

The text views of Association Rules generated through RapidMiner is shown in figure 2.



```
[6260136920021, 6260074050026] --> [6261715202347] (confidence: 0.974)
[6260281414093] --> [6260608980317, 6261715202347] (confidence: 0.974)
[6260608980317, 6261715202347, 6260281411061] --> [6260136920021] (confidence: 0.974)
[6260608980317, 6260281411061] --> [6260136920021] (confidence: 0.974)
[6260281411061, 2001726000037, 2001726000020] --> [6260608980317, 6261715202347] (confidence: 0.974)
[6260608980317, 6261715202347] --> [6260136920021] (confidence: 0.974)
[6260608980317] --> [6260136920021] (confidence: 0.974)
[6260281411061, 2001804000010] --> [6261715202347] (confidence: 0.974)
[6261715202347, 6260281411061] --> [6260136920021] (confidence: 0.974)
[6260281411061] --> [6260136920021] (confidence: 0.974)
[6261715202347] --> [6260136920021] (confidence: 0.974)
[6260136920021, 6260281411061, 2001804000010] --> [6261715202347] (confidence: 0.974)
[6260136920021, 2001726000037, 2001726000020] --> [6260608980317, 6261715202347] (confidence: 0.974)
[6260136920021] --> [6260608980317, 6261715202347] (confidence: 0.974)
[2001804000010] --> [6261715202347] (confidence: 0.974)
[6260136920021, 6260074023181] --> [6260608980317, 6261715202347] (confidence: 0.974)
[6260136920021, 2001804000010] --> [6261715202347] (confidence: 0.974)
```

Fig 2: Association Rules generated through RapidMiner

Some samples of association rules generated by RapidMIner are shown in Figure 2.The minimum confidence was specified as 80%. RapidMiner generates all possible association rules from the dataset. And each rule has a specific means, For example the first rule from figure 2 means the one who bought the items with their representative barcodes (6260608980317&6260074050026) with 0.974 percentage of confidence will buy another item (6261715202347). The number of generated rules, support and confidence may be specified by providing the corresponding parameters.

6. Conclusion

Since almost all mid to large size retailers today possess electronic sales transaction systems, retailers realize that competitive advantage will no longer be achieved by the mere use of these systems for purposes of inventory management or facilitating customer checkout. In contrast, Use of an association rule mining driven application to manage retail businesses will provide retailers with reports regarding prediction of product sales trends and customer behavior. This will allow retailers to make hands-on, knowledge-driven decisions and competitive advantage will be gained by those retailers who are able to extract the knowledge hidden in the data, generated by those systems, and use it to optimize their marketing decision making.

Corresponding Author:
Afshin Mirzaei
Department of Industrial Engineering and Management, Shahrood University of Technology,
Shahrood, Iran
E-mail: mirzaei.afshin@gmail.com

References

1. Blischok, T. Every transaction tells a story. In Chain Store Age Executive with Shopping Center Age, 71 (3),50-57, 1995
2. Borin, N. and Farris, P.W, A sensitivity analysis of retailer shelf management models. Journal of Retailing, 1995, 71(2), pp.153–171.
3. Yang, M-H., and Chen, W.-C., A study on shelf space allocation and management, International Journal of Production Economics, 1999, pp 60–61, 309–317.
4. R. Agrawal and R. Srikant, "Fast Algorithms for Mining Association Rules", Proceedings of the 20th VLDB Conference Santiago, Chile, 1994.
5. R. Agrawal, T. Imielinski, A. Swami, Mining association rules between sets of items in large databases, Proceedings of the ACM SIGMOD International Conference on Management of Data, 1993, pp. 207–216.
6. J Han, J Pei ,Y Yin, "Mining frequent patterns without candidate generation," Proc ACM SIGMOD, Dallas, TX, USA, ACM Press May 2000, Vol. 29, No. 2, pp. 1-12.
7. Osmar R. ZaÃ³ane, Mohammad El-Hajj, and Paul Lu, "Fast parallel association rule mining without candidacy generation,"First IEEE International Conference on Data Mining, San

Jose, California, USA, IEEE CS Press, November 2001, pp.665-668.

8. Li Liu, Eric Li, Yimin Zhang, and Zhizhong Tang, "Optimization of frequent itemset mining on multiple-core processor," 33rd International Conference on Very Large Data Bases, Vienna, Austria, VLDB Endowment Press, September 2007, pp. 1275-1285.

9. R. Agrawal, T. Imielinski, and A.N. Swami, "Mining association rules between sets of items in large databases," Proc. of the 1993,ACM SIGMOD International Conference on Management of Data, Washington, D.C., USA, ACM Press, May 1993, Vol. 22, No. 2,pp. 207 216.

10. Mannila, H. 1997. Methods and problems in data mining. In Afrati, F.; and Kolaitis, P. eds. Proceedings of the International Conference on Database Theory, Springer-Verlag,41-55.

11. Agrawal, R.; Mannila, H.; Srikant, R.; Toivonen, H.; and Verkamo, A. 1996. Fast discovery of association rules. In Fayyad, U.; Piatetsky Shapiro, G.; Smyth, P.; and Uthurusamy, R., eds. Advances in Knowledge Discovery and Data Mining. Menlo Park, CA: AAAI Press. 307-328.

12. R. Agrawal and R. Srikant, "Fast algorithms for association rules," Proceedings of the 20th International Conference on VLDB, Santiago,USA, Morgan Kufmann Press, September 1994. pp. 487– 499.

7/22/2012