Journal of King Saud University – Computer and Information Sciences (2013) **25**, 77–87

King Saud University
Journal of King Saud University – Computer and Information Sciences
www.ksu.edu.sa
www.sciencedirect.com

ORIGINAL ARTICLE

An approach to products placement in supermarkets using PrefixSpan algorithm

George Aloysius *, D. Binu

Research & Development, Griantek, Bengaluru, India

Received 18 July 2011; revised 14 April 2012; accepted 7 July 2012
Available online 25 July 2012

KEYWORDS
Sequential pattern mining;
Supermarkets;
PrefixSpan;
Product placement;
Retailing

Abstract With a great variation of products and user buying behaviors, shelf on which products are being displayed is one of the most important resources in retail environment. Retailers can not only increase their profit but, also decrease cost by proper management of shelf space allocation and products display. To solve this problem, we propose an approach to mine user buying patterns using PrefixSpan algorithm and place the products on shelves based on the order of mined purchasing patterns. The proposed approach is able to mine the patterns in two stages of process. In the first stage, the sequences of product categories are mined to place the product categories on the shelves based on the sequence order of mined patterns. Subsequently, in the second stage, the patterns (products) are mined for each category and then, rearrange the products within the category by incorporating the profit measure on the mined patterns. The experimentation is carried out on the synthetic datasets and the evaluation with two datasets showed that the proposed approach is good for product placement in supermarkets.

1. Introduction

Due to the recent competition in the retailing industry, retailers are striving to improve their revenue in order to run their stores more efficiently. The term *competition* in retail is the rivalry between retailers who are keen to obtain the same customer. To improve sales and revenue, various analyses are performed by a retailer to determine which different products should be merchandized together based on the historic purchasing behavior. Recent marketing research has suggested that in-store environmental stimuli, such as shelf-space allocation and product display, have a great influence upon consumer buying behavior and may induce substantial demand (Chen et al., 2006) . Shelf-space allocation and product display are the problems of efficiently arranging retail products on shelves in order to maximize profit, improve stock control, improve customer satisfaction, etc. (Silva et al., 2009).

Nowadays, different displaying strategies directly influence customer's purchasing decision and profit of retail stores. Product placement is an amazing bit of marketing science. The strategy *Product to shelf placement* is a critical retailing problem having major impact on the financial performance of retail stores. Managing this problem successfully will obviously result in overall retail store's profit. Therefore, the decision-making process regarding this problem should be integrated to increase the retailer profitability. The decision

* Corresponding author.
E-mail addresses: aloysius_g@griantek.com (G. Aloysius), binu@griantek.com (D. Binu).
Peer review under responsibility of King Saud University

Production and hosting by Elsevier

1319-1578
http://dx.doi.org/10.1016/j.jksuci.2012.07.001

to stock products among the large number of competing products and the placement of those products on shelves is a central question of retailing. In other words, the problem is to decide which combination of products to be displayed on which shelves that have the greatest value to customers so as to maximize the store's total profit. A nice displaying method of products, not only attracts the sight and attention of consumers, but also increases extra consumption and customer satisfaction. Therefore, how to appropriately allocate product items on suitable shelf space becomes a very important issue in retailing business (Tsai and Wu, 2010).

The location of a product in a shop can make a crucial difference to its sales. So, we have to know the information about the products, which are often bought together and this information can be effectively used in this analysis to determine which products should be placed next to each other. This business process known as, market basket analysis is one of the most prominent applications of data mining. Here, the data mining techniques are used to find the pairs, triples of product sequences which are bought together. Such a set of products is called frequent itemsets and the order of frequent items, called sequences. Recently, researchers have acknowledged that a frequent sequential pattern (Hou and Zhang, 2008; Masseglia et al., 2003; Agrawal and Srikant, 1995; Parmar and Garg, 2007; Sobh, 2007; Zhao and Bhowmick, 2003; Agrawal and Srikant, 1994; Antunes and Oliveira, 2003; Pei et al., 2004; Pei et al., 2001; Yukhuu et al., 2008) is the best measure that can be employed to decide '*which combination of products should be placed close together on which positions of which shelves?*' in supermarkets. Sequential patterns can assist managers to determine which items are bought one after other in a cycle, or to examine the placement of products on shelves by considering the profit of retail stores and more.

In this paper, we have developed a sequential pattern mining approach for product placement problem in supermarkets. In the proposed approach, the sequential patterns are mined from the transaction database into two stages: In the first stage, the category database is created by replacing the items in the transaction database with their corresponding categories specified in the category table. Subsequently, the product categories are mined from the history purchasing patterns in the category database using projection-based sequential pattern mining algorithm and then, the product categories are placed on shelves according to the sequential order. And, in the second stage, the category-transaction database is constructed for all categories in the transaction database and the sequential patterns (products) within the category-transaction database are mined by the PrefixSpan algorithm. After that, the profit measure is incorporated into the mined sequential patterns to determine the most profitable item and to place the products within the category.

The remainder of the paper is organized as follows: In Section 2, a brief review of research related to the proposed approach is given. In Section 3, the sequential pattern mining problem is defined, and the PrefixSpan algorithm is presented. Our proposed approach using PrefixSpan for product placement in supermarket is explained with examples in Section 4 and the experimental results and performance analysis are discussed in Section 5. Section 6 provides the practical implications and future direction of the research. Finally, our approach is concluded in Section 7.

2. Review of related works

Literature presents a few of research relevant to product placement and shelf allocation problem. Here, we review seven different techniques available in the literature. Brijs et al. (2004) have integrated the discovery of frequent itemsets with a (microeconomic) model for product selection (PROFSET). The model enabled the integration of both quantitative and qualitative (domain knowledge) criteria. Furthermore, they demonstrated that the impact of product assortment decisions on overall assortment profitability can easily be evaluated by means of sensitivity analysis. On the other hand, Chen et al. (2005) have integrated customer behavioral variables, demographic variables, and transaction database to establish a method of mining changes in customer behavior. The approach for mining changes in customer behavior can assist managers in developing better marketing strategies.

Chen et al. (2006) have used data mining techniques to discover the implicit, meaningful, relationship between the relative spatial distance of displayed products and the items' unit sales in a retailer's store. They presented a representation scheme and developed a robust algorithm based on association analysis. To show its efficiency and effectiveness, an intensive experimental study using self-defined simulation data was conducted. Similar to Chen et al. (2006) and Chen and Lin (2007) have utilized a popular data mining approach, association rule mining, instead of space elasticity to resolve the product assortment and allocation problems in retailing. They have applied multi-level association rule mining to explore the relationships between products as well as between product categories.

The specific problem of how to allocate a fixed amount of shelf space to different products within a particular product category was addressed by Reyes and Frazier(2007). A nonlinear integer goal programing formulation was proposed to consider both profitability and customer service factors. The decision support tool was shown that the tradeoffs between increased profitability and improved customer service allowed the manager to make the best tradeoff for the situation. Nafari and Shahrabi (2010) have developed an approach to optimally select and price the products and allocate them to shelf space with consideration of their prices. The paper has taken advantage of data mining techniques, association rules, to find relationships between products regarding their prices. Finally, to show the efficiency and effectiveness of the approach, the experiment on real world data was executed.

Application of data mining techniques in library data results in interesting and useful patterns that can be used to improve services in University libraries. Sitanggang et al. (2010) have presented the results of the work in applying the sequential pattern mining algorithm namely, AprioriAll on a library transaction dataset. Frequent sequential patterns containing book sequences borrowed by students were generated for minimum supports of 0.3, 0.2, 0.15 and 0.1. These patterns helped to develop the library in providing book recommendation to students, conducting book procurement based on readers' need, as well as managing books' layout.

By analyzing the above discussed works, the technique given in Brijs et al. (2004) described about selecting the product and the work given in Chen et al. (2005) and Reyes and Frazier, 2007 discussed about developing the marketing strategies using the patterns mined. Importantly, the techniques presented in Chen et al. (2006), Nafari and Shahrabi (2010) and Chen and Lin (2007) are taken for the product allocation problem that usually happened in supermarket. An interesting work was described in Sitanggang et al. (2010) that provided a technique to book recommendation using the rules mined. These works are real motivation of our research in developing the strategy for product placement. Here, we have used the sequential patterns mined from the database for product placement so that the sequence buying behavior will motivate the customers to buy the nearly located products.

3. Problem statement

This section presents a basic description of sequential pattern mining which, was first introduced in Agrawal and Srikant (1995) and extended in Srikant et al. (1996). In addition, a detailed description of PrefixSpan algorithm is given, which is a prominent algorithm for mining sequential patterns and also, a brief explanation of supermarket model for the product placement problem is given.

3.1. Sequential pattern mining

The sequential pattern mining problem is to mine a complete set of sequential patterns with respect to a given sequence database, D_S and a support threshold, min_sup is stated as follows: Let D_s be a sequential database where, each transaction T contains customer-id, and a set of items involved in the transaction. Let $I = \{i_1, i_2, \ldots, i_m\}$ be a unique set of items. An itemset is a non-empty subset of items, and an itemset with k items is called a k-itemset. A sequence S is an ordered list of itemsets based on the time stamp. It is denoted as $< s_1, s_2 \ldots, s_n >$, where $s_j, j \in 1, 2, \ldots, n$ is an itemset which, is also called as an element of the sequence S and $s_j \subseteq I$. A sequence of k items (or of length k) is called k-sequence. For example, $<(1)(3)(5)>$, $<(2)(3,4)>$ and $<(1)(2)(1)>$ are all 3-sequences. A sequence $<s_1, s_2, \ldots, s_n>$ is called a sub-sequence of another sequence $< s'_1, s'_2 \ldots, s'_q >$, $(n \leqslant q)$ if there exist an integer $1 \leqslant i_1 \leqslant i_2 \leqslant \cdots, i_n \leqslant q$ such that $s_1 \subseteq s'_{i_1}$, $s_2 \subseteq s'_{i_2}, \ldots, s_n \subseteq s'_{i_n}$. For instance, $<(2)(5)>$ is a subsequence of $<(4)(2)(1)(3,5)>$ since $(2) \subseteq (2)$ and $(5) \subseteq (3,5)$ .The support of sequence S in a sequence database D_S is the number of transactions that contained the sequence S. The sequence S is called a frequent sequential pattern in the sequential database such that $\sup(S) \geq \min\_\sup$ where, min_sup is a given positive integer, support threshold. The problem of sequential pattern mining is to find a complete set of frequent sequential patterns satisfying a minimum support in the sequence database. A sequential pattern with length l is called an l-pattern. (Pei et al., 2004).

3.2. Prefixspan algorithm

At first, we introduce some basic definitions used in the PrefixSpan algorithm.

Definition 1 (*Prefix*). All the items present in an element are listed in an alphabetical order. A sequence $\beta = \langle e'_1 e'_2 \ldots e'_m \rangle$ $(m \leqslant n)$ is called a prefix of the given sequence $\alpha = \langle e_1 e_2 \ldots e_n \rangle$ (where each e_i corresponds to a frequent element in S), if and only if (1) $e'_i = e_i$ *for* $(i \leqslant m-1)$; (2) $e'_m \subseteq e_m$; and (3) all the frequent items in $(e_m - e'_m)$ are subsequent to those in e'_m.

Definition 2 (*Suffix*). Let $\beta = \langle e_1 e_2 \ldots e_{m-1} e'_m \rangle$ $(m \leqslant n)$ be the prefix of a given sequence $\alpha = \langle e_{1,} e_{2,} \ldots, e_n \rangle$. The sequence $\gamma = \langle e''_m e_{m+1} \ldots e_n \rangle$ is said to be the suffix of α with regard to prefix β, denoted as $\gamma = \alpha/\beta$ or $\alpha = \beta \cdot \gamma$, where $e''_m = (e_m - e'_m)^2$. Note, the suffix of α with regard to β is empty if β is not a subsequence of α.

Definition 3 (*Projected database*). If α is a sequential pattern of a given sequence database S, then the $\alpha - projected$ database is defined as the collection of suffixes of sequences in S with regard to prefix α, denoted as $S|_{\alpha'}$.

Definition 4. Support count in projected databaseLet α be a sequential pattern in sequence database S, and β be a sequence with prefix α, then the support count of β in $\alpha - projected$ database $S|_{\alpha'}$ is the number of sequences γ in $S|_{\alpha'}$ such that $\beta \subseteq \alpha \cdot \gamma$, denoted as $\text{support}_{S|_\alpha}(\beta)$.

The execution procedure of PrefixSpan algorithm is described as follows: At first, the PrefixSpan algorithm scans the sequential database to find all frequent items which are 1-length sequential patterns with respect to the given minimum support. Then, the sequence database is partitioned into different subsets according to these frequent items, where each subset is the projection of the sequence database with respect to the frequent 1-sequence. The subsets of sequential patterns are mined by constructing the corresponding set of projected databases. From these projected databases, the PrefixSpan algorithm continues to form the frequent 2-sequences with the same corresponding prefix using frequent 1-sequences. For every frequent k-sequence, the PrefixSpan algorithm generates a projected database to mine the frequent (k + 1)-sequences recursively. Based on these definitions, the algorithm of the PrefixSpan is presented in Fig. 1.

4. An approach for placing products on shelves in supermarkets using Prefixspan algorithm

Today, the retailing sector in the economy is an extremely competitive arena. Retailers are keen to do everything possible to make their systems more efficient, while maximizing their profit. Several tactics are used to influence consumers' purchases, including product assortment (deciding which merchandise to sell), store layout and space planning, merchandise pricing, services offered, advertising and other promotional programs. Among these, store layout and product placement (deciding which combination of products to locate on shelves) planning focuses on the improvement of the visual effect of the shopping environment and also the sales productivity.

However, unplanned (occasional) purchases are very common. An attractive display of the products based on customers' purchasing pattern could increase impulse purchases and also it makes the customer feel good about the shop. Previous research shows that unplanned purchases make up about one third of all transactions in many retail stores. Therefore, product to shelf placement is an area worthy of investigation in which retailers have the opportunity to increase their sales. However, placing of hundreds or even thousands of products is challenging. On one hand, products have different categories and product types which is not an easy task for retailers. Due to this complex task, they would prefer to increase the profit by

Input : A sequence database D_S, and the minimum support threshold min_sup

Output: The complete set of sequential patterns

Method: Call *PrefixSpan* $(\langle\ \rangle, 0, D_S)$.

Subroutine: *PrefixSpan* $(\alpha, l, D_S|_\alpha)$

Parameters: α: a sequential pattern; l: the length of α; $D_S|_\alpha$: the α-projected database, if $\alpha \neq \langle\ \rangle$; otherwise, the sequence database D_S.

Method:

1. Scan $D_S|_\alpha$ once, find the set of frequent items I_f such that
   a) I_f can be assembled to the last element of α to form a sequential pattern; or
   b) $\langle I_f \rangle$ can be appended to α to form a sequential pattern.
2. For each frequent item I_f, append it to α to form a sequential pattern α', and output α';
3. For each α', construct α'-projected database $D_p|_{\alpha'}$, and call *PrefixSpan* $(\alpha', l+1, D_S|_{\alpha'})$.

Figure 1 The PrefixSpan algorithm.

placing the products into their corresponding categories based on the customer purchasing pattern.

4.1. A Supermarket layout model: an overview

Supermarket is a large self-service grocery store, selling a wide variety of food, groceries and dairy products and household goods. It is larger in size and has a wider selection than a traditional grocery store, also selling items typically found in a convenience store. The supermarket typically comprises of meat, fresh products and baked goods categories, along with shelf space reserved for canned and packaged goods as well as for various non-food items such as household cleaners, pharmacy products and pet supplies. Most supermarkets are similar in design and layout due to the trend in marketing. Fresh products tend to be located near the entrance of the store. Milk, bread, and other essential staple items are usually situated toward the rear of the store and in other out-of-the-way places, purposely done to maximize the customer's time spent in the store, strolling past other items and capitalizing on impulse buying. The front of the store, or "front end"' is the area where, the point of sale machines or cash registers are usually located. Many retailers also have implemented self-checkout devices in an attempt to reduce labor costs. For example, a sample layout design for simple supermarket is given in Fig. 2.

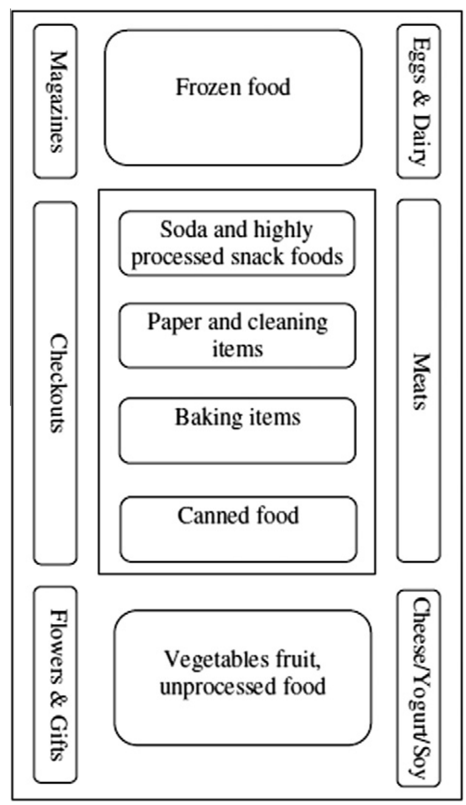

Figure 2 Supermarket layout.

In general, the take-over of food retailing has occurred much more rapidly in processed, dry, and packaged foods such as noodles, milk products and grains, for which supermarkets have an advantage over traditional stores due to economies of scale. Usually, the supermarkets' progress in gaining control of fresh food markets has been slower and the first fresh food category for the supermarkets to gain a majority share include "commodities" such as potatoes, and sectors experiencing consolidation in first-stage processing and production: often chicken, beef and pork, sea foods and fish (Reardon et al., 2004). For recognizing the importance of selling food and household goods in order to attract customers, the modern supermarkets have made significant progress in improving their supply and display of products. For example, the hierarchical structure of product categories and products within the supermarket is shown in Fig. 3.

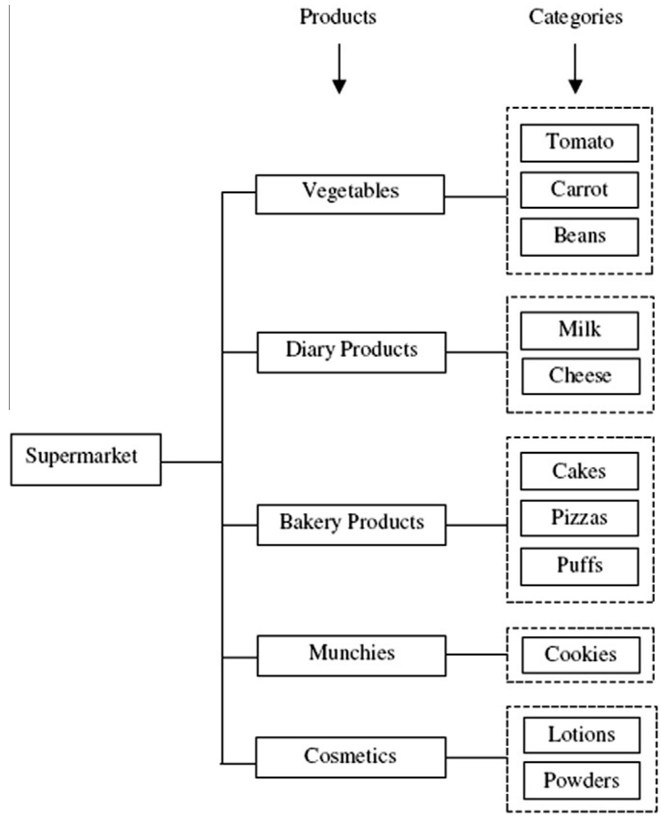

Figure 3 Hierarchical structure of the supermarket products.

4.2. Prefixspan algorithm for product placement

In general, there has been a trend in supermarkets to improve the sales in order to display the products efficiently. The product to shelf placement problem is a real-world problem faced by many retail companies. The problem involves in placing the customer required combination of products among different product categories held within a retail store. To solve this problem, we have proposed an approach for mining of user's buying patterns from trade history databases using PrefixSpan mining algorithm. Based on the mined sequential patterns, the products are placed on shelves to increase the sales of the supermarkets. The proposed approach is designed to mine the patterns from trade history database into two stages such as, (1) Mine the sequential patterns among different product categories, and (2) Mine the patterns of products for each product category associated with profit. In the first stage, PrefixSpan algorithm mines the sequential patterns (product categories) from the history transaction database with item categories and then, the mined product categories are placed on shelves in sequential order. And, in the second stage again, we use the PrefixSpan algorithm for each category to mine the sequential patterns within the category and place the products that are generally in sequential order. In addition, the profit is incorporated for the mined patterns which can assist the retailer to find out which products are more significant in the category.

Let D_T be a history transaction database containing a set of transactions where, each transaction has a set of items. Each transaction in the database D_T includes n number of different products (items) based on their respective categories, $T = \{C_1, C_2, C_3, \ldots, C_n\}$. Therefore, we have a category table CT, which gives the information about which products belong to which category. In addition to this, we have a profit table, PT which is used to denote which products have more significance among different products (items) in the database D_T .The profit table is used in the second level of our proposed approach. The detailed implementation of our proposed approach for product placement using PrefixSpan algorithm is described with example in the following subsections.

Example. Let us consider the transaction database D_T given in Table 1 with min_sup = 2 and the category table CT corresponding to the transaction database D_T given in Table 2. Also, the profit table is given in Table 3. The sequential patterns S can be mined by a prefix-projection method in two stages and are described as follows.

4.2.1. Mine the sequential patterns among different product categories

In this stage, the PrefixSpan algorithm is used to mine the sequences of product categories and then the resultant product categories are placed on the shelves based on the sequential order of mined patterns. For this, we have to construct the category database using the transaction database and the category table and then, the sequential patterns are mined from this database by using PrefixSpan algorithm. It has the following steps which, are described as follows:

Step 1: Constructing category database for the items within the transaction database

Table 1 Transaction database.

Transaction ID	Items				
1	b	e	f		
2	a	b	d	g	
3	a	d	g		
4	b	c	e		
5	a	b	d	e	f

Table 2 Category table.

Category ID	Items			
C_1	a	b		
C_2	c	d	f	G
C_3	e			

Table 3 Profit table.

Items	Profit
a	73
b	114
c	93
d	37
e	59
f	70
g	115

At first, we replace the items in transaction database by their corresponding categories i.e., replacing the items by respective category ids to build a category database. Then, the sequential patterns are mined from the constructed category database by PrefixSpan algorithm.

Example. The given transaction database D_T contains five transactions with items and also, the category table has three categories, each having different distinct items such as $C_1 = \{a, b\}$, $C_2 = \{c, d, f, g\}$ and $C_3 = \{e\}$. Now, we replace the items $\{a, b\}$ by C_1, $\{c, d, f, g\}$ by C_2 and $\{e\}$ by C_3 to obtain the category table. The resultant category table after replacing the items by their categories is shown in Table 4.

Step 2: Generating a set of 1-length sequential patterns

Here, we scan the database D_T once to mine all frequent 1-length sequential patterns that satisfy the predefined minimum support.

Example. In this example, the database is scanned once to find all the 1-length patterns that satisfy the min_sup . Based on the min _sup , we obtain the frequent 1-length patterns that are, $<C_1>:5$, $<C_2>:5$ and $<C_3>:3$ where, the notation '$<$pattern$>$:support' represents the frequent pattern and its support value.

Step 3: Constructing projection Databases for 1-length sequential patterns

The 1-length sequential patterns mined in the previous step are used to construct the projected database defined in definition 1. The number of projection databases is based on the number of mined frequent 1-length sequential patterns. By

Table 4 Category Database.

Transaction ID	Items Corresponding Category				
1	C_1	C_3	C_2		
2	C_1	C_1	C_2	C_2	
3	C_1	C_2	C_2		
4	C_1	C_2	C_3		
5	C_1	C_1	C_2	C_3	C_2

using these complete set of 1-length frequent patterns, we can obtain k disjoint subsets of projection databases from the sequential database if the mined sequence contains k number of patterns.

Example. Here, we form three projected databases for the length-1 frequent patterns such as $\{C_1, C_2$ and $C_3\}$. The projected database for the pattern $<C_1>$ is constructed as follows: Considering the first transaction in the database, the projection based on this transaction is obtained by taking the postfixes of pattern $<C_1>$ (sequences after its first occurrence) in the first transaction. Similarly, the projections of the other transactions are obtained to complete the projection database for the pattern $<C_1>$. The projected database for the pattern $<C_1>$ contains, $<C_3, C_2>$, $<C_1, C_2, C_2>$, $<C_2, C_2>$, $<C_2, C_3>$ and $<C_1, C_2, C_3, C_2>$. In the similar way, the projection database for the other two patterns $<C_2>$ and $<C_3>$ is constructed. The projected database of all 1-length patterns in the projection set is shown in Table 5.

Step 4: Finding subsets of sequential patterns

In this step, we scan the projected database once to mine a set of 2-length frequent patterns that satisfy the minimum support. Then, the projected database is constructed again for the obtained 2-length frequent patterns and this process is repeated recursively until all patterns are mined.

Example. For mining all 2-length sequential patterns, the projected database formed by the 1-length sequential patterns is used. The mining of 2-length sequential patterns having prefix $<C_1>$ is described as: First, we scan the projected database once to obtain the count of frequent items represented as, $[<C_1>:2$, $<C_2>:5$ and $<C_3>:3]$ and then the mined 2-length sequential pattern is the combination of all the three patterns that satisfy the minimum support. Next, the 3-length patterns are obtained by forming the projected database again based on the 2-length sequential patterns and scan the projected database once. Recursively, we mine all the length patterns with prefix $<C_1>$. The aforementioned procedure is repeated for other 1-length patterns $<C_2>$ and $<C_3>$. The mined all length patterns are listed in Table 6.

Table 5 Projected Databases.

Prefix	Projected (suffix) database
$<C_1>$	$<C_3, C_2>$ $<C_1, C_2, C_2>$ $<C_2, C_2>$ $<C_2, ;C_3>$ $<C_1, C_2, C_3, C_2>$
$<C_2>$	$<C_2>$ $<C_2>$ $<C_3>$ $<C_3, C_2>$
$<C_3>$	$<C_2>$ $<C_2>$

Table 6 Projected Databases and Sequential Patterns.

Prefix	Projected (suffix) database	Sequential patterns
$<C_1>$	$<C_3, C_2>$ $<C_1, C_2, C_2>$ $<C_2, C_2>$ $<C_2, C_3>$ $<C_1, C_2, C_3, C>$	$<C_1>$, $<C_1C_1>$, $<C_1C_2>$, $<C_1C_3>$, $<C_1C_1C_2>$, $<C_1C_2C_3>$, $<C_1C_3C_2>$ $<C_1C_2C_2>$, $<C_1C_1C_2C_2>$
$<C_2>$	$<C_2>$ $<C_2>$ $<C_3>$ $<C_3, C_2>$	$<C_2>$, $<C_2C_2>$, $<C_2C_3>$
$<C_3>$	$<C_2>$ $<C_2>$	$<C_3>$, $<C_3\ C_2>$

4.2.2. Mining patterns of products for each product category associated with profit

Here, we mine the patterns for each category in the transaction database by using PrefixSpan algorithm. After that, we use a weight measure (profit) to identify the important or high profit items (products) within the sequential pattern. Based on the importance, the products are placed within the category of the supermarket to increase the sales. Most of the previous sequential pattern mining algorithms, sequential patterns and items within sequential patterns have been treated uniformly, but real sequences differ in their importance. For this reason, the weight measure has been suggested here to efficiently place the products on shelves. Normally, the attribute values of items such as, price (profit) can be used as a weight measure for market basket data. Here, we are using the profit table which gives different weights (profits) to items within the sequential patterns of the transaction database. Hence, we construct the *category-transaction* database for all categories in the transaction database at first and then we mine the patterns from each *category-transaction* database using the PrefixSpan algorithm. Then, we use the profit measure for the mined sequential patterns to identify the most profitable product and then, we place the products based on their profit values. This procedure is implemented by using the following steps described as follows:

Step 1: Constructing category-transaction database for all categories

In this step, the *category-transaction* databases for n categories are constructed by dividing the transaction database into n based on the number of categories present in the transaction database. The *category-transaction* includes all transactions where, each transaction contains only the items of the respective category itself.

Example. Let us consider the above example, we have three categories of products $\{C_1, C_2$ and $C_3\}$ as shown in Table 2. Therefore, the transaction database in Table 1 is divided into three *category-transaction* databases shown in Tables 7–9.

Step 2: Mining of patterns from each category-transaction database

After the product placement in category level, the second stage is to decide the product placement in product type level in the same category. In this stage, the PrefixSpan algorithm is used again to mine the sequential patterns from the category-transaction database. Hence, the steps (from step 2 to step 4) described in the above subsection are utilized to mine the sequential patterns in all *category-transaction* databases.

Example. Using PrefixSpan algorithm, we mine for each category-transaction database to mine the frequent patterns (products) within the category. The mined patterns for each category are listed in Table 10. The item $<c>$ in category C_2 is an infrequent pattern since it has a support less than the minimum support.

Table 7 Category-transaction databases for C_1.

Transaction ID	Items	
1	b	–
2	a	b
3++	a	–
4	b	–
5	a	b

Table 8 Category-transaction databases for C_2.

Transaction ID	Items	
1	f	–
2	d	g
3	d	g
4	c	–
5	d	f

Table 9 Category-transaction databases for C_3.

Transaction ID	Items
1	E
2	–
3	–
4	E
5	E

Step 3: Using profit measure for the mined sequential patterns

In this step, the profit measure is utilized for mined sequential patterns to determine the most profitable items and then, the sequential order of patterns is rearranged based on the profit value. Generally, profit measure plays an important role in retail environments. Here, we utilize this profit measure to give precise results for product placement problem. The profited sequential pattern (PSP) value of the pattern X is calculated as:

$$PSP(X) = \frac{\sup(X) * profit(X)}{l(X)}$$

where,

$sup(X) \rightarrow$ Support value of pattern X in database D_T
$profit(X) \rightarrow$ Profit value of pattern
X $l(X) \rightarrow$ Length of pattern X

Table 10 Sequential patterns for category-transaction database.

Category-transaction database	Patterns
C_1	<b>, <a>, <ab>
C_2	<f>, <d>, <dg>, <g>
C_3	<e>

Example. In this example, the profited sequential patterns are obtained for all patterns of all categories. The profited sequential patterns obtained for category C_2 are {<g>, <dg>, <f>, <d>}. Here, the pattern <g> has high profit and the pattern <d> has low profit.

4.2.3. Transforming the mined patterns to relevant location in supermarket

Once we mine the sequential patterns, the finding of relevant location in supermarket is an important step in the proposed technique. Suppose, if we obtain a sequential pattern from product category database like, {<g>, <dg>, <f>}, then these three categories of products should be placed in upper rock (<g>), middle rock (<dg>) and lower rock (<f>). Based on the rocks available in the supermarket layout, the arrangement of product type product may be slightly varied. In the second case, every product sequence falling within the same category should be placed in single rock so that the product can be easily visible to the customer.

5. Results and discussion

In this section, the experimental results of the proposed approach for product placement by effectual mining of sequential patterns using PrefixSpan algorithm is described. The proposed approach has been programed using JAVA (jdk 1.6) and the experimentation is performed on a 3.0 GHz Pentium PC machine with 2 GB main memory.

5.1. Sample experimental results

For experimentation, a sample database, category table and a profit table are given as like, in Tables 1–3 respectively. In our proposed approach the patterns are mined into two stages by using PrefixSpan algorithm. In the first stage, the category database was constructed for the input database and then, we mined sequential patterns from the category database based on the minimum support given as min_sup = 2. In the second stage, we constructed a category-transaction database for all categories in the transaction database and the patterns were mined from each category-transaction database .After that, the profit measure is incorporated for the mined patterns to identify the most profitable item and the patterns are rearranged based on the profit value. Finally, we placed the products on shelves based on the sequence of mined patterns. Here, the order of the patterns is the most important factor for placing products on shelves in supermarkets.

The performance of the profit associated PrefixSpan is compared with the PrefixSpan in order to evaluate the quality of our approach. The comparative result clearly ensures that the proposed approach provides optimal order of sequential patterns compared to PrefixSpan algorithm. The proposed algorithm generates sequential patterns that contain only the profitable order of sequential patterns (for example, <f>, <d>, <dg>, <g>) but, the PrefixSpan algorithm contains all the less profitable order of sequential patterns (<g>, <dg>, <f>, <d>). Hence from a business perspective, profit associated PrefixSpan algorithm is more suitable for developing better display strategies of products in supermarkets compared to PrefixSpan algorithm. The comparison result is given in Table 11.

Table 11 Comparison result of PrefixSpan and profit associated PrefixSpan.

Category-transaction database	PrefixSpan patterns	PrefixSpan patterns with profit
C_1	<b>, <a>, <ab>	<b>, <a>, <ab>
C_2	<f>, <d>, <dg>, <g>	<g>, <dg>, <f>, <d>
C_3	<e>	<e>

As shown in Fig. 4, the products and their categories are placed on shelves based on the mined patterns to provide the hierarchical structure of the supermarket. Here, the mined categories are placed in the first level of hierarchical structure and then in the second level, the products are placed within each category based on the mined patterns. It ensures that our approach efficiently solves the product placement problem and also enhances the visual appearance of the stores to increase the sales.

5.2. Experimentation with large data

The experimentation of the proposed approach is performed using a large dataset that contains 50,000 transactions of 50 items. Here, we have considered two datasets for experimentation, (i) Rule generation dataset (ii) Validation dataset. In rule generation, the significant sequential patterns are mined to find the layout of product mapping. After that, the mined patterns are validated with the second datasets whether the mined sequence is presented there. Additionally, the second dataset is obtained from the supermarket that is based on the layout given by the first dataset. If most of the patterns are frequent in the second dataset and also, the frequency is improved compared with the first dataset, this signifies that the proposed product mapping provides the better significance in supermarket about improving the revenue of a corresponding supermarket.

5.2.1. Experimental results

The experimental results obtained from the proposed approach with the largest dataset are described in this sub-section. Initially, the input dataset is given to the proposed approach to mine the sequent category so that the category products can be placed together. The mining performance with respect to the time and number of categories mined is given in the following graphs shown in Figs. 5 and 6. From the graph, we can identify that the time required to mine the patterns from the transaction database of size 50,000 is higher than the database size of 40,000. One more thing, the time needed to find the pattern greatly depends on the support we have provided. If the given support value is equal to 50, the time

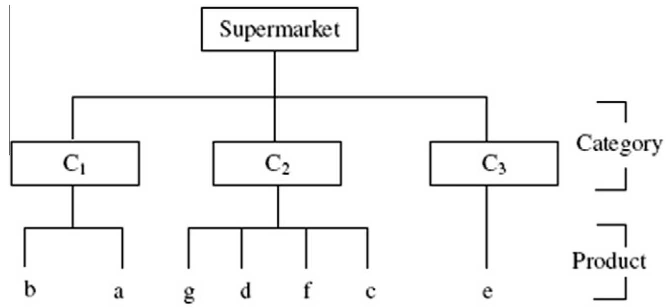

Figure 4 Hierarchical structure of product mapping in supermarkets.

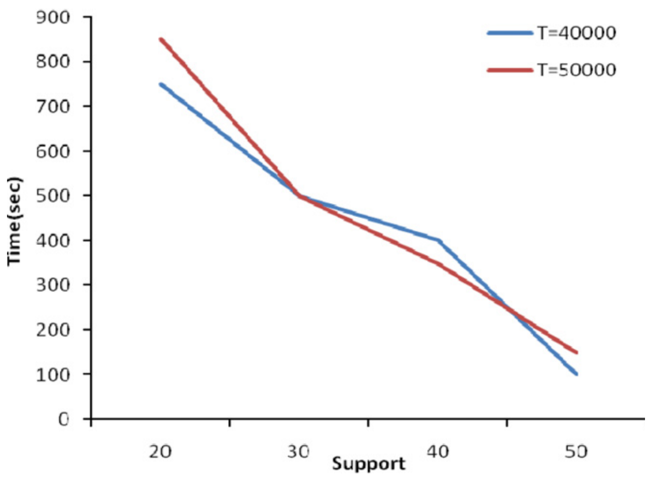

Figure 5 Time complexity.

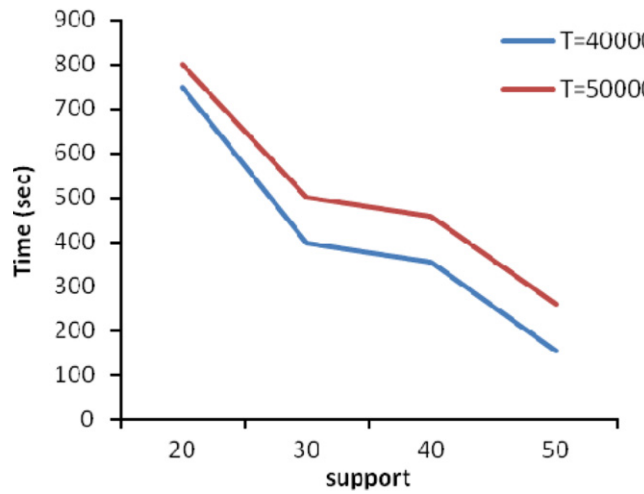

Figure 7 Time complexity.

complexity is 100 s despite, the time complexity is 750 s for the support value equivalent to 20. After analyzing the number of patterns generated, two different sizes of transactions are generated with the same number of patterns (450) for the support value equivalent to 45. In the similar way, for every category, the sequential patterns are mined so that the neighbor sequence can be found out. The obtained results are plotted in the graphs that are shown in Figs. 7 and 8. From the graph shown in Fig. 7, the computation time of two different sizes follow the same path for various support size given as user input. The number of patterns mined for various supports of the category database does not follow the meaningful path. Because, some variance is there for various supports.

5.2.2. Experimental evaluation

This section describes the experimental evaluation of the proposed approach in product mapping. The evaluation is difficult to this type of problem because, after finding the layout from

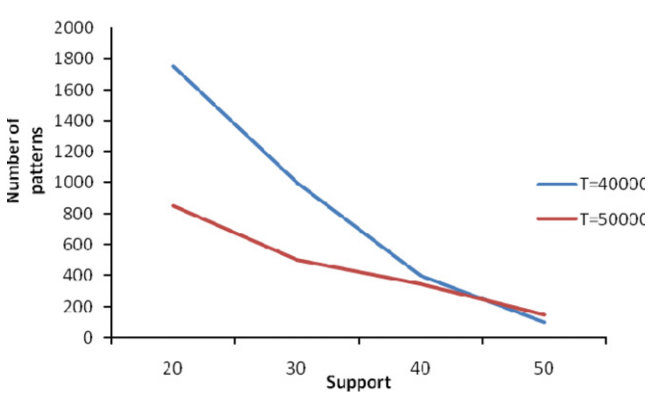

Figure 6 Number of categories mined.

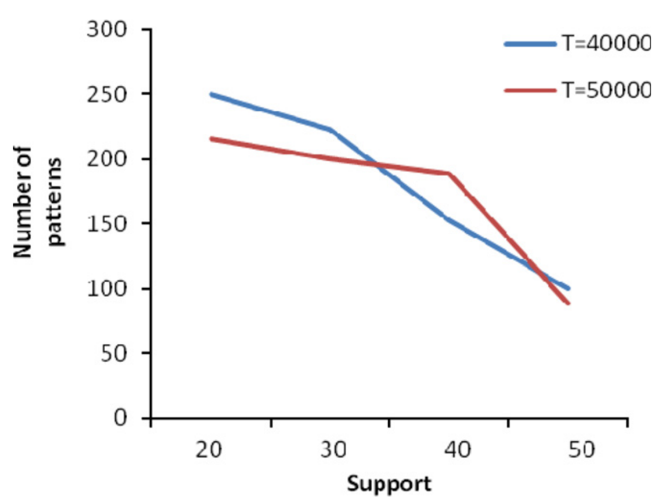

Figure 8 Number of patterns mined.

Table 12 Evaluation report of the proposed approach.

	First dataset		Second dataset	
	Number of similar patterns mined	Average support	Number of similar patterns mined	Average support
1-length	20	50	20	53
2-length	80	42	80	45
3-length	58	38	58	40
4-length	42	30	42	38

the mined patterns, the same layout should be fixed in the supermarket. Then, the items moved should be analyzed so that only the performance of the proposed approach can be analyzed. Through this idea, the patterns mined from the sample data are formed as hierarchical architecture described in Fig. 4. Then, the frequency of buying the sequent product is compared with the existing layout about the performance. But, this is an expensive and risky way of comparing the proposed approach in product mapping. Here, the mined patterns are validated with other similar datasets whether they are frequent in the dataset. If this is frequent, we can justify that the different layout provide the same sequence that surely affect the customer behavior if the sequences of product are put together.

With the same perspective, two similar datasets are taken and the layout patterns are generated from the first dataset. Then, the second dataset is given to the mining procedure such a way; the similar patterns mined from these two datasets are evaluated. Actually, the second dataset is synthetically generated with the view of installation which will be done in respect to the first patterns. The support of the mined patterns from first dataset is relatively high in the second dataset as per the table given in Table 12. From the table, we can identify that the average support of 1-length patterns are increased from 50 to 53. When we are looking at other length patterns, we can see the increasing support in all the different length patterns, signifying the importance of product placements in supermarket database.

6. Practical implications and future direction

There has been a wide opportunity to make more revenue from the supermarket business if they started to implement the proper product placement. The product placement problem surely brings the more revenue for the business people because the neighbor product placement should surely attract the customers to buy the product. So, the revenue and optimization management need this type of technique to bring even more business by simply ordering the products in the proper way. The same problem can be applicable to various business areas where the customer wants to buy the product after seeing it. And also, the retailing and stock industry will get more benefit because the allocation is a crucial problem there.

The same technique can be extended including the concept of revenue optimization to make the same technique even more business perspective. Various optimization techniques can be used to find the most suitable patterns since the proposed technique mine large number of sequential patterns. Another way, the strategy discussed here is to make the customers to buy some products by locating it in neighbor place. But, if we put the strategy of ‘‘must buy together’’, there may be a chance to buy the product that they have not pre-planned. Accordingly the technique can be extended for the strategy of ‘‘must buy together’’ so that the comparative study can be possible in between two techniques in the perspective of profit in supermarket. Also, in future, the real data will be used to analyze the performance of the proposed approach.

7. Conclusion

In this paper, we have presented an efficient approach, for placing products on shelves in supermarkets by using PrefixSpan algorithm, which mine all sequential patterns from the customer transaction database. As a result, the products are allocated on shelves based on these sequential orders of mined patterns. The PrefixSpan algorithm employed a pattern-growth methodology that finds sequential patterns in two stages: In the first stage, PrefixSpan algorithm used to mine the sequences of product categories and then the product categories are placed on shelves based on the sequence order of mined patterns. Subsequently, in the second stage again the patterns (products) are mined for each category using PrefixSpan algorithm and then rearrange the products within the category by incorporating the profit measure on the mined patterns. The experimentation is carried out on the synthetic datasets and the experimental results showed that the proposed approach is effective for better product placement in supermarkets.

References

Agrawal, R., Srikant, R. 1994. Fast algorithms for mining association rules. In: Proceedings of the International Conference on Very Large Databases (VLDB’94), Santiago, Chile, pp. 487–499.

Agrawal, R., Srikant, R. 1995. Mining sequential patterns. In: Proceedings of the 11th International Conference on Data Engineering, Taipei, Taiwan, pp. 3–14.

Antunes, Cláudia, Oliveira, Arlindo L., 2003. Generalization of pattern-growth methods for sequential pattern mining with gap measures. Machine Learning and Data Mining in Pattern Recognition. Lecture Notes in Computer Science, vol. 2734. Springer, pp. 239–251.

Brijs, Tom, Swinnen, Gilbert, Vanhoof, Koen, Wets, Geert, 2004. Building an association rules framework to improve product assortment decisions. Data Mining and Knowledge Discovery 8 (1), 7–23.

Chen, Mu-Chen, Lin, Chia-Ping, 2007. A data mining approach to product assortment and shelf space allocation. Expert Systems with Applications 32 (4), 976–986.

Chen, Mu-Chen, Chiu, Ai-Lun, Chang, Hsu-Hwa, 2005. Mining changes in customer behavior in retail marketing. Expert Systems with Applications 28 (4), 773–781.

Chen, Yen-Liang, Chen, Jen-Ming, Tung, Ching-Wen, 2006. A data mining approach for retail knowledge discovery with consideration of the effect of shelf-space adjacency on sales. Decision Support Systems 42 (3), 1503–1520.

Sizu Hou, Xianfei Zhang. 2008. Alarms association rules based on sequential pattern mining algorithm. In: Proceedings of the Fifth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '08), vol. 2, Shandong, October 2008, pp. 556–560.

Masseglia, Florent, Poncelet, Pascal, Teisseire, Maguelonne, 2003. Incremental mining of sequential patterns in large databases. Data and Knowledge Engineering 46 (1), 97–121.

Nafari, Maryam, Shahrabi, Jamal, 2010. A temporal data mining approach for shelf-space allocation with consideration of product price. Expert Systems with Applications 37 (6), 4066–4072.

Parmar, Jatin D., Garg, Sanjay, 2007. Modified web access pattern (mWAP) approach for sequential pattern mining. Journal of Computer Science 6 (2), 46–54.

Jian Pei, Jiawei Han, Behzad Mortazavi-asl, Helen Pinto, Qiming Chen, Umeshwar Dayal, Mei-chun Hsu, 2001. PrefixSpan: mining sequential patterns efficiently by prefix-projected pattern growth. In: Proceedings of the 17th International Conference on Data Engineering (ICDE'01), Heidelberg, Germany, April 2001, pp. 215–224.

Pei, Jian, Han, Jiawei, Mortazavi-Asl, Behzad, Wang, Jianyong, Pinto, Helen, Chen, Qiming, Dayal, Umeshwar, Hsu, Mei-Chun, 2004. Mining sequential patterns by pattern-growth: the PrefixSpan approach. IEEE Transactions on Knowledge and Data Engineering 16 (11), 1424–1440.

Reardon, Thomas, Timmer, Peter, Berdegue, Julio, 2004. The rapid rise of supermarkets in developing countries: induced organizational, institutional, and technological change in agrifood systems. E-Journal of Agricultural and Development Economics 1 (2), 168–183.

Reyes, Pedro M., Frazier, Gregory V., 2007. Goal programming model for grocery shelf space allocation. European Journal of Operational Research 181 (2), 634–644.

Dario Landa Silva, Fathima Marikar, Khoi Le. 2009. Heuristic approach for automated shelf space allocation. In: Proceedings of the 24th ACM Symposium on Applied, Computing (SAC'09), pp. 922–928.

Sitanggang, I.S., Husin, N.A., Agustina, A., Mahmoodian, N. 2010. Sequential pattern mining on library transaction data. In: International Symposium in Information Technology, vol. 1, Kuala Lumpur, June 2010, pp. 1–4.

Sobh, Tarek, 2007. Innovations and Advanced Techniques in Computer and Information Sciences. Springer, ISBN 978-1-4020-6268-1.

Srikant, R., Agrawal, R. 1996. Mining sequential patterns: generalizations and performance improvements. In: Proceedings of the 5th International Conference on Extending Database Technology (EDBT'96), Avignon, France, September 1996, pp. 3–17.

Chieh-Yuan, Tsai, Ming-Chang, Wu. 2010. Applying a two-stage simulated annealing algorithm for shelf space allocation problems. In: Proceedings of the World Congress on Engineering (WCE'10), June 30 – July 2, London, U.K., pp. 2376–2380.

Yukhuu, Ankhbayar, Garamragcha, Sansarbold Hwang Young Sup, 2008. Mining sequential patterns by Prefixspan algorithm with approximation. In: Proceedings of the 8th WSEAS International Conference on Applied Computer Science (Acs'08).

Qiankun Zhao, Sourav S. Bhowmick. 2003. Sequential pattern mining: a survey. Technical Report, CAIS, Nanyang Technological University, Singapore, No.118.