A Multi-Data Mining Approach for Shelf Space Optimization
Considering Customer Behaviour

Sheng-Hsiang Huang[1], Chieh-Yuan Tsai[1] and Chih-Chung Lo[2]
[1]*Department of Industrial Engineering and Management, Yuan Ze University, Taoyuan, Taiwan*
[2]*Department of Applied Informatics, Fo Guang University, Yilan County, Taiwan*
cytsai@yzu.edu.tw, locc@mail.fgu.edu.tw

Keywords: Data Mining, Shelf Space Assignment, Moving and Purchase Patterns, Customer Behaviour.

Abstract: A well product-to-shelf assignment strategy can help customers easily find product items and dramatically increase the retailing store profit. Previous studies in this area usually applied the space elasticity to optimize product assortment and space allocation models. However, a well product-to-shelf assignment strategy should not only consider product assortment and space elasticity. Thus, this study develops a product-to-shelf assignment approach by considering both product association rules and traveling behaviour of consumer. Specifically, the first task of this research is to develop a method to discover traveling behaviour of consumer, which includes both product association rules and traveling behaviour of consumer, in the store. The second task is to construct and solve a product-to-shelf assignment model, based on the information provided in the first task. In this research, products are classified as major item, minor item and the others. Only minor will be reassigned. Experimental result shows our proposed method can reassign minor items to suitable shelves and increase cross-selling opportunity of major and minor items.

1 INTRODUCTION

To implement a retail strategy, store managers should develop a retail mix that satisfies the need of its target market. The elements in retail mix include store location, product assortment, pricing, advertising and promotion, store design and display, services and personal selling (Levy and Weitz, 1995). Among that, product assortment and shelf space allocation are two important issues which dramatically affect customers' purchasing decisions (Yang, 1999). However, using space elasticity for shelf space allocation needs to estimate a great quantity of parameters which results in high cost and errors in the mathematical model.

Recently, the progress of information technology makes retailers easily collect daily transaction data. With the novel information technology, retailers can solidify ephemeral relationships with customers into long-term and fruitful relationships if they can discover customer behavior from collected data. Data mining is one of the most popular technologies that discover potential customer knowledge from business databases to assist a policy decision. Chen and Lin (2007) applied the multi-level association rule mining to explore the relationships between products as well as between product categories for resolving the product assortment and allocation problems in retailing. Although the association rules to assist managers in developing better layout for stores, their method is more suitable for the case of new stores or joint sales. For an existing store, the frequent purchase pattern may not maintain if the customer's interesting products are not at the original locations or shelves anymore.

Except purchasing association between products, customer traveling behaviors/patterns should be considered in solving product-to-shelf assignment problem. When shopping in a store, a customer travels around the aisles of a store, stops at certain locations, deliberates about his/her consideration, and chooses the best options. This process is repeated until the whole shopping trip completes. Recently, some studies tried to tackle the shopping path problem. Larson et al. (2005) presented exploratory analyses of an extraordinary new dataset that reveals the path taken by individual shoppers around an actual grocery store, as provided by RFID tags located on their shopping carts.

As mentioned above, some researchers employed product association rules mining to improve shelf space allocation, while other researchers focused on how to derive shopping paths of consumers. However, to maximize the cross-selling possibility for retail stores, product association and traveling patterns should be integrated and considered at the same time when dealing with shelf space management. This paper, therefore, solves the product-to-shelf assignment problem by taking both product association rules and traveling patterns of consumers into consideration. By combining moving logs and payment records, customer mobile transaction sequences (i.e., sequences of moving path with purchased transactions) can be used to represent the customer purchasing behavior in detail (Yun and Chen, 2007). Furthermore, valuable behavior patterns should be discovered to reflect the actual profit of product items, utility mining can find the patterns not only with high appearing frequency but with high utility values (Shie et al., 2012). Specifically, the first task of this research is to develop a method to discover traveling behavior of consumer, which includes both high-utility mobile sequential patterns and product association rules, in the store. The second one is to construct a complete model, based on the information of the first task, for solving product-to-shelf assignment problem.

The remainder of this paper is organized as follows. Section 2 formally defines the research problem and introduces important components of the proposed method such as store layout, association rule, high-utility mobile sequential pattern mining, and product assignment procedure. In Section 3, an empirical performance evaluation is conducted. Finally, conclusion is summarized in Section 4.

2 THE PROPOSED METHOD

The framework of the proposed method consists of three main stages as shown in Figure 1. The main task in stage 1 is to collect required data. When the customer completes the shopping, purchasing transaction recording customer id, purchased items and quantities are stored into the transaction database. Next, the traversal path represented by readers is retrieved from the traversal-temp database and transformed into a traversal sequence represented by sections. Finally, the system will combine purchasing transactions and traversal sequence of the customer as a mobile transaction sequence and store it into the mobile transaction sequence database. The system architecture in the first stage of the proposed method is shown in Figure 2.

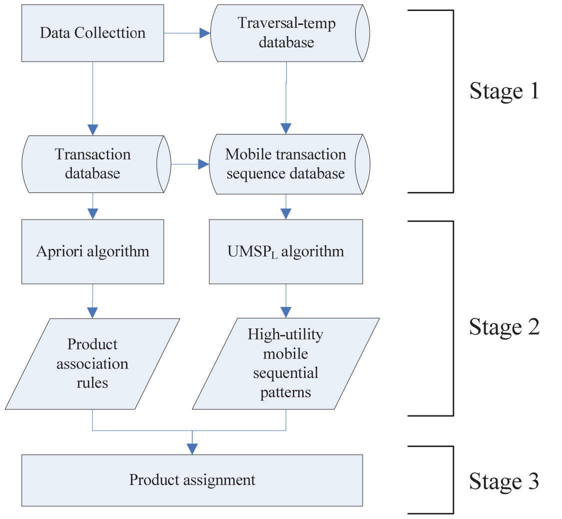

Figure 1: The framework of the proposed product assignment method.

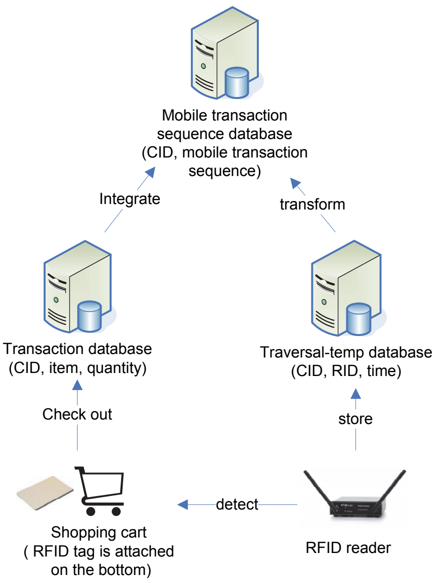

Figure 2: Information architecture for collecting behaviour data.

Typically, customers tend to follow certain sequential patterns to pick up their required items. For example, a customer might go to section A and take nail polish first, then move to section C and take pants, and move to section F and take diamond rings before check out. Although previous studies did apply different sequential pattern mining approaches to obtain frequent sequential patterns, these patterns tends to prefer frequent patterns instead of valuable patterns. Thus, the major task in the second stage is to explore customers' high-utility mobile sequential patterns based on the mobile transaction sequence database and the utility of items. The high-utility mobile sequential pattern is the sequential pattern containing a list of frequent visiting sections and frequent purchased items at the corresponding paths. Meanwhile, another major task in the second stage is to derive the product association rules from the transaction database.

In the third stage, based on the product association rules and high-utility mobile sequential patterns generated in stage 2, a three-step product assignment method is proposed to rearrange items into suitable shelves. The first step is to find all items ever appeared in the high-utility mobile sequential patterns. These items are defined as "major item". The second step is to find minor items related with major items from the product association rules. The final step is to rearrange minor items to suitable shelves based on the information derived from high-utility mobile sequential patterns and product association rules.

2.1 Store Layout

The gray oval in Figure 3 indicates the coverage area of an RFID reader. Let $I = \{i_1, i_2, \ldots, i_g\}$ be the set of all product items sold in the store, $Z = \{z_1, z_2, \ldots, z_y\}$ be the set of all shelves in the store, $S = \{S_1, S_2, \ldots, S_n\}$ be the set of sections in the store, and $R = \{R_1, R_2, \ldots, R_m\}$ be the set of RFID readers.

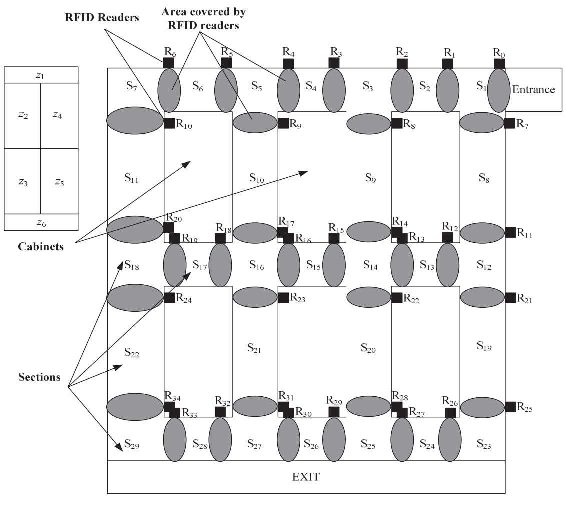

Figure 3: The layout of an example store.

With the purchasing transaction and traversal sequence, a mobile transaction sequence of a customer can be derived and defined as $TS = <T_1, T_2, \ldots, T_n>$ where transaction $T_j = (Se_j; \{[it_{j1}, q_{j1}], [it_{j2}, q_{j2}], \ldots, [it_{jh}, q_{jh}]\})$ represents that a customer purchases $\{[it_{j1}, q_{j1}], [it_{j2}, q_{j2}], \ldots, [it_{jh}, q_{jh}]\}$ in section Se_j where q_{jp} is the purchased quantity of pth item $it_{jp} \in I$ in transaction T_j. A path is denoted as $Se_1Se_2 \ldots Se_r$, where $Se_j \in S$ and $1 \le j \le r$. All the mobile transaction sequence (*TS*) will be stored in the mobile transaction sequence database (*TSD*).

2.2 Association Rules and Mining

The association rule mining is to discover the rules of the presence of one set of items implies the presence of another set of items from a given transaction database. The form of rule can be represented as $X \rightarrow Y$ where X and Y are the antecedent and consequent of the rules respectively.

The Apriori algorithm, one of the most popular methods for frequent pattern mining introduced by (Agawal and Srikant, 1994), is adopted in this research to get item associations from transaction database. In the algorithm, L_k denotes the set of all frequent k-itemset and C_k denotes the set of candidate k-itemset. In this research, only "single item to singe item" rules are needed. Therefore, this research will stop at L_2 in the Apriori algorithm and then generate rules based on provided minimum confidence. All association rules then are stored in product association rule database (*ARD*).

2.3 High-Utility Mobile Sequential Patterns and Mining

The $UMSP_L$ (high Utility Mobile Sequential Pattern by a Level-wised method) algorithm proposed by Shie et al. (2012) is adopted in this paper to obtain high-utility mobile sequential patterns. The $UMSP_L$ algorithm consists of four steps. The inputs of the $UMSP_L$ algorithm include a mobile transaction sequence database (*TSD*), a pre-defined utility table, a minimum support threshold (δ), and a minimum utility threshold (ε). The first three steps are to find *WUMSPs* based on the sequence weighted downward closure (SWDC) property (Liu et al., 2005), while the forth step is to find high-utility mobile sequential patterns (*UMSPs*). In step 1, the mobile transaction sequence database (*TSD*) is scanned several times to generating all *WULIs* (high sequence weighted utilization section-itemset) and each *WULI* is mapped to a specific identity in a mapping table. Note that the mapped *WULIs* are 1-*WULPs* (high sequence weighted utilization section-pattern). In step 2, the mobile transaction sequence database (*TSD*) is transformed into a trimmed database (*TD*) by mapping the *WULIs* to their new identities. The section-items which are impossible to be the elements of high-utility mobile sequential patterns are removed from the database. In step 3, the trimmed database (*TD*) is utilized to find the *WUMSPs* (high sequence weighted utilization mobile sequential pattern) by the proposed level-wised method. This step is the key to mining performance and its procedure is shown in Figure 4. In step 4, the *WUMSPs* are checked to find *UMSPs* (high-utility mobile sequential patterns) by an additional scan of the mobile transaction sequence database (*TSD*). The *WUMSPs* whose utilities are larger than or equal to ε are regarded as high-utility mobile sequential patterns. All *UMSPs* are then stored in high-utility mobile sequential pattern database (*SPD*).

Input: All 1-*WULPs*, a trimmed database *TD*, a minimum support threshold δ, and a minimum utility threshold ε
Output: *WUMSPs*
Join the *1-WULPs* to form candidate 2-*WULPs* and then store them into 2-candidate trees;
For each candidate 2-*WULP* X_T
Perform an additional scan of *TD*;
If $sup(X_T) \geq \delta$ and $SWU(X_T) \geq \varepsilon$
X_T is a 2-*WULP*;
End If
Next
Generate 2-*WUMSPs* by joining the 2-*WULPs* with their corresponding paths in the 2-candidate trees;
$k = 3$;
While (candidate *WULP* is generated)
Generate candidate *k-WULPs* by combining the (*k*-1)-*WULPs* of the two (*k*-1)-*WUMSPs* whose paths are equal to each other;
Store the generated candidate *k-WULPs* into *k*-candidate trees;
For each candidate *k-WULP* X_T
Perform an additional scan of *TD*;
If $sup(X_T) \geq \delta$ and $SWU(X_T) \geq \varepsilon$
X_T is a *k-WULP*;
End If
Next
Generate *k-WUMSPs* by joining the *k-WULPs* with their corresponding paths in the *k*-candidate trees;
$k = k + 1$;
End While

Figure 4: Third step of the $\mathrm{UMSP_L}$ algorithm.

Let's take the following simple example to explain the computation process of the $UMSP_L$ algorithm. Assume the minimum support threshold δ is 2 and the minimum utility threshold ε is 100. In addition, the utility table and trimmed database *TD* is shown in Table 1 and Table 2, respectively. In *TD*, the original mobile transaction sequences are parsed into the sequences of section-itemsets and paths. For instance, < S_4; t_2; 3> in customer CID 1' means that t_2 occurred in S_4, and S_4 is in the third position of the path. Note that if there is no item in the start or end location of a path, the location in a path will be trimmed.

Table 1: Utility table.

Item	Profit ($ per unit)	Item	Profit ($ per unit)
i_5	20	i_{43}	10
i_{11}	5	i_{50}	12
i_{22}	6	i_{58}	8
i_{24}	15	i_{62}	6
i_{38}	8		

Table 2: Transformed mobile transaction sequence database *TD*.

CID	Sequence of $WULI_S$	Path	SU
1'	<S_2; t_1; 1>, <S_4; t_2; 3>, <S_{20}; t_8; 7>	$S_2S_3S_4S_3S_9S_{14}S_{20}$	93
2'	<S_8; t_3, t_4, t_{10}; 1>, <S_{13}; t_5; 3>, <S_{15}; t_6; 5>, <S_{21}; t_9; 7>	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{21}$	153
3'	<S_8; t_3; 1>, <S_{13}; t_5; 3>, <S_{15}; t_6; 5>, <S_{17}; t_7; 7>, <S_{21}; t_9; 9>	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{17}S_{16}S_{21}$	160
4'	<S_2; t_1; 1>, <S_4; t_2; 3>, <S_{15}; t_6; 7>, <S_{20}; t_8; 9>	$S_2S_3S_4S_5S_{10}S_{16}S_{15}S_{14}S_{20}$	89
5'	<S_8; t_3, t_4, t_{10}; 1>, <S_{13}; t_5; 3>, <S_{15}; t_6; 5>, <S_{21}; t_9; 7>	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{21}$	134

In the third step, the candidate 2-*WULPs* are generated by joining the 1-*WULPs* in the mapping table, and the result is stored into *k*-candidate trees (*k* is the length of the patterns). Each *k*-candidate tree stores the candidate *k-WULPs* whose last section-itemsets are the same. After constructing 2-candidate trees, an additional scan of *TD* is performed to check the path support and *SWU* of each candidate *2-WULP* and to form the paths in the moving patterns. After generating 2-*WUMSPs*, candidate 3-*WULPs* are generated by combining the 2-*WULPs* of two 2-*WUMSPs* if the path of one 2-*WUMSP* contains the path of another 2-*WUMSP*. The processes will be recursively executed until no further candidate moving pattern is generated. In this example, 2-candidate tree and 4-candidate tree with root of <S_{21}; t_9> are respectively shown in Figures 5(a) and 5(b). Figure 5(a) indicates five 2-*WUMSPs* marked with solid lines are generated, while Figure 5(b) shows three 4-*WUMSPs* are obtained. In the fourth step, after all *WUMSPs* are generated, an additional scan of the database is performed to check for real high utility mobile sequential patterns. The *WUMSPs* whose utilities are greater than or equal to the minimum utility threshold are regarded as high utility mobile sequential patterns. For example, five 2-*UMSPs*, seven 3-*UMSPs*, and three 4-*UMSPs* of $<S_{21}; t_9>$ are found.

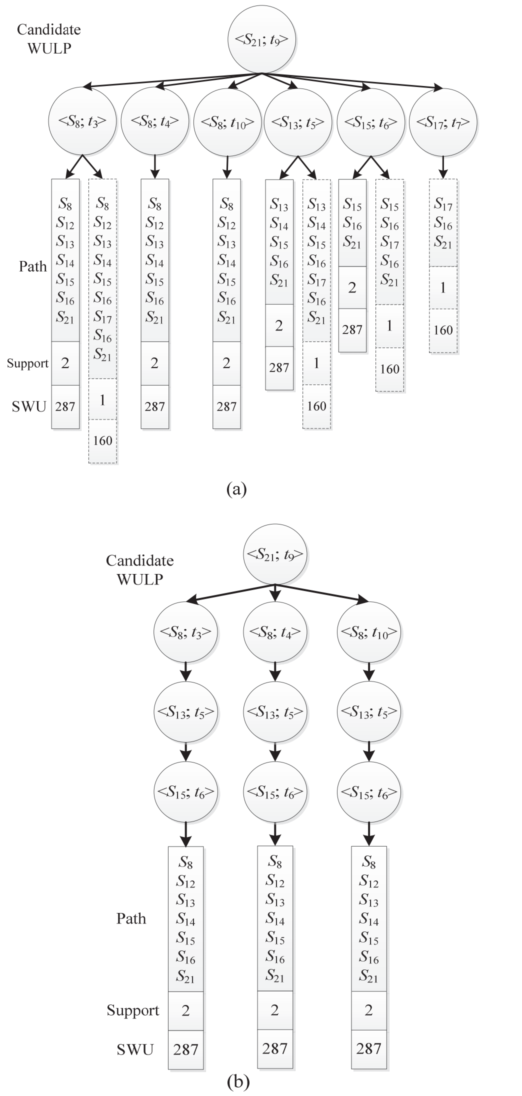

Figure 5: (a) 2-candidate tree of $<S_{21}; t_9>$; (b) 4-candidate tree of $<S_{21}; t_9>$.

2.4 Item Classification

In this study, an item in the store will be classified as *major item* or *minor item* based on the following definitions.

Definition 1: a *major item* is the item ever appeared in high-utility mobile sequential patterns. Major items are considered as important commodities attracting customers to purchase. If major items are reassigned to other section(s)/shelves(s), the high-utility mobile sequential patterns might be invalid since the major attractions are not at the original place anymore. Therefore, major items are considered as the items not been rearranged. In the following discussion, the set of major items is denoted as *MA*.

Definition 2: a *minor item* is the item allowed to be reassigned. Minor items are considered as affiliated commodities related to major items. Minor items can be found according to the following rules. First, all association rules in product association rule database (*ARD*) are checked. If the item on the consequent of an association rule is a major item, the item on the antecedent of the association rule is a candidate minor item. Next, if the candidate minor item is not a major item, the item will be a minor item and added into the set of minor items *MI*.

2.5 Product Reassignment

According to definitions 1 and 2, major items should not be rearranged because frequent customer visiting behaviours will not maintain if major items are not displayed at their original positions. Therefore, only minor items will be rearranged.

To increase the cross sale possibility of minor items, minor items should be rearranged to the location as close as possible to its related major items according to previous customer visiting and purchasing behaviors. Therefore, based on the product association rule database and high-utility mobile sequential pattern database, this study develops an Item Location Preference Evaluation (ILPE) procedure to calculate location preference if a minor item is placed at a section in the store.

First, for each minor item mi_j in *MI*, the procedure scans product association rule database (*ARD*) and retrieve all major items in the consequent of a rule while the antecedent of the rule is mi_j. The set of major items related to mi_j is denoted as GM_j. Then, for each major item ma_k in GM_j, the procedure will scan high-utility mobile sequential pattern database (*SPD*) and find out the set of high-utility mobile sequential patterns containing ma_k, which is denoted as GP_{jk}. For each high-utility mobile sequential pattern $UMSP_m$ in GP_{jk}, the procedure will evaluate the *movement distance* that minor item mi_j is assigned to section s_n. Let $D_{k,m}^{j,n}$ be the movement distance in $UMSP_m$ if mi_j is moved from the section that major item ma_k is located at to section s_n.

If no relationship among minor item mi_j, major item ma_k, section s_n, and high-utility mobile sequential pattern $UMSP_m$ can be found, $D_{k,m}^{j,n}$ is set as β. Notes that β is the threshold of maximum section movement and is provided by users.

If $D_{k,m}^{j,n}$ is close to 0, minor item mi_j should have high possibility to be rearranged to section s_n. Therefore, the standardization of assigning mi_j to s_n under the condition of high-utility mobile sequential pattern $UMSP_m$ and major item ma_k is defined as:

$$W_{k,m}^{j,n} = \frac{D_{k,m}^{j,n}}{\beta}, \qquad 0 \le D_{k,m}^{j,n} \le \beta \tag{1}$$

where β is the threshold of maximum section movement and $0 \le W_{k,m}^{j,n} \le 1$.

The input of the procedure is high-utility mobile sequential patterns database (*SPD*), products association rule database (*ARD*), major item set (*MA*) and minor item set (*MI*), while the output is the item location preference matrix $[f_{j,y}]$.

Note that it is assumed that every product item in this research has the same size so that two minor items in different shelves can be exchanged directly. After that, this paper will try to reassign products to most suitable shelves based on information of matrix $[f_{j,y}]$. The objective of product rearrangement is to rearrange minor items and keeps the numbers of section movement as less as possible. Hungarian method (Kuhn, 1955) is adopted in this study. The Hungarian method is a combinatorial optimization algorithm that can solve the assignment problem.

3 IMPLEMENTATION AND EXPERIMENTAL RESULTS

3.1 Data Description

A simplified supermarket as illustrated in Figure 6 is used to demonstrate the feasibility of the proposed shelf space allocation method. The supermarket is divided into 37 sections (s_1 to s_{37}) and 52 shelves (z_1 to z_{52}) according the instruction mentioned in Section 2.1. Customers enter the supermarket from entrance s_1 and check out their purchase from section s_{32} or section s_{37}. There are 119 product items in this store in which an item belongs to one of the 16 product classes.

However, the RFID system is not deployed in this example store right now. Thus, a mobile transaction sequence generator is developed to simulate the shopping behaviors in the supermarket. In this study, the total number of mobile transaction sequences in the generator is set as 1,000. With the mobile transaction sequences, the transaction of each customer can be obtained.

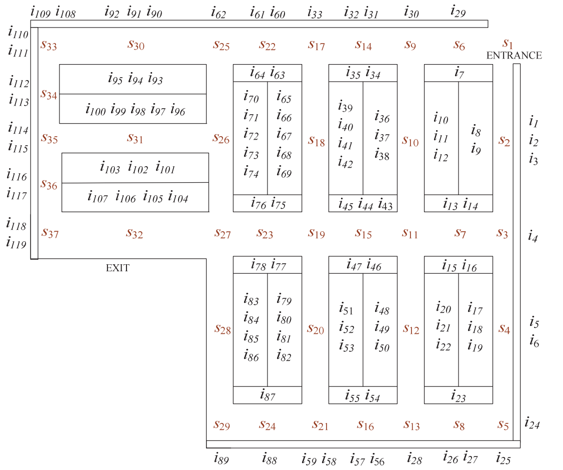

Figure 6: Physical location of all products.

3.2 Experimental Results

Based on the transactions from the generator, 24 product association rules are generated using Apriori algorithm when minimum support = 10% and minimum confidence = 60%. Part of product association rules is illustrated in Table 3. Next, $UMSP_L$ algorithm is applied to generate high-utility mobile sequential patterns based on the utility data in Table 4. There are 16 high-utility mobile sequential patterns generated when minimum support count is 6 and the minimum utility is 150. Part of the high-utility mobile sequential patterns is shown in Table 5.

Table 3: Association rule (10%, 60%).

ID	Rule	ID	Rule	ID	Rule
1	$i_4 \rightarrow i_{43}$	…	…	24	$i_{105} \rightarrow i_{62}$

Table 4: The utility table for items.

Item	Profit	Item	Profit	Item	Profit
i_1	100	…	…	i_{119}	20

Table 5: High-utility mobile sequential pattern (minimum support count=6).

PID	Pattern
1	$<\{<S_3; i_4><S_{15}; i_{43}><S_{25}; i_{62}><S_{32}; i_{105}>\}; S_1, S_2, S_3, S_{11}, S_{15}, S_{19}, S_{25}, S_{32}>$
…	…
16	$<\{<S_{18}; i_{65}><S_{21}; i_{58}>\}; S_1, S_6, S_9, S_{14}, S_{17}, S_{18}, S_{19}, S_{20}, S_{21}>$

After association rules and high-utility mobile sequential patterns are generated, major items and minor items can be found. In this simulation, 9 major items including i_4, i_{30}, i_{43}, i_{48}, i_{58}, i_{62}, i_{65}, i_{83} and i_{105} are found. In addition, 7 minor items including i_8, i_{22}, i_{34}, i_{42}, i_{76}, i_{80} and i_{107} are identified. With major and minor items, β is set as 3, the location preference weight matrix $[f_{j,y}]$ can be obtained according to Equation (1).

The final stage of the proposed shelf space allocation method is to reassign minor items to their best shelf location using Hungarian method. Table 6 shows the reassignment result after taking Hungarian method. We find that i_8 is strongly related to major item i_{48}, i_{42} is strongly related to major item i_{30}, i_{76} is strongly related to major item i_4, and i_{107} is strongly related to major item i_{43}. Thus, the four minor items (i_8, i_{42} , i_{76} , i_{107}) are re-organized to the location close to their major items (i_{48} , i_{30} , i_4 , i_{43}). Minor item i_{80} does not change shelf location since i_{80} already located on the shelf very close to its major item i_{58} at the original layout. Minor item i_{34} is not assigned to the best shelf z_{36} since the location preference weights are calculated based on average concept.

Table 6: Result of assignment.

Minor Item	Original Shelf	New Shelf
i_8	z_8	z_{13}
i_{22}	z_{13}	z_{46}
i_{34}	z_{19}	z_{21}
i_{42}	z_{21}	z_{19}
i_{76}	z_{34}	z_8
i_{80}	z_{36}	z_{36}
i_{107}	z_{46}	z_{34}

4 CONCLUSIONS

In retailing business, a well product-to-shelf assignment strategy will affect customers' purchasing decision and increase profit for a retailing store. Thus, this research proposes a novel method for product-to-shelf assignment taking both frequent purchased product relationship and shopping path knowledge into considerations. With the proposed method, market managers can generate a better products' layout. Our method determines major items and minor items before product reassignment. Instead of reassigning all of items in the store, this research reassigns minor items only. As mentioned, this research trends to rearrange products depend on information of product's relationship and utility, and customer's shopping path.

ACKNOWLEDGEMENTS

This work was partially supported by the National Science Council, Taiwan, R.O.C. under No. NSC 102-2221-E-431-002

REFERENCES

Agrawal, R., Srikant R., 1994. Fast algorithms for mining association rules', *Proceedings of the 20th International Conference on Very Large Data Bases*, VLDB, 487-499, Santiago, Chile.

Chen, M.C., C.P. Lin, 2007. A data mining approach to product assortment and shelf space allocation. *Expert Systems with Applications*, 32 (4), 976-986.

Kuhn, H.W., 1955. The Hungarian Method for the assignment problem. *Naval Research Logistics Quarterly*, 2, 83-97.

Larson, J.S., Bradlow, E.T., Fader, P.S., 2005. An exploratory look at supermarket shopping paths. *International Journal of Research in Marketing*, 22 (4), 395-414.

Levy, M. and Weitz, B.A., 1995. *Retailing management*, Irwin, Chicago.

Liu, Y., Liao, W.K., Choudhary, A., 2005. A fast high utility itemsets mining algorithm. *Proceedings of the 1st International Workshop on Utility-Based Data Mining*, 90-99.

Shie, B.E., Hsiao, H.F., Tseng, V.S., 2012. Efficient algorithms for discovering high utility user behavior patterns in mobile commerce environments. *Knowledge Information Systems*, 37 (2), 363-387.

Yang, M.H., 1999. An efficient algorithm to allocate shelf space. *European Journal of Operational Research*, 131(1), 107-118.

Yun, C.H. Chen, M.S., 2007. Mining mobile sequential patterns in a mobile commerce environment. *IEEE Transactions on Systems, Man and Cybernetics-Part C*, 30 (2), 278-295.