

Integrating Product Association Rules and Customer Moving Sequential Patterns for Product-to-Shelf Optimization

Chieh-Yuan Tsai and Sheng-Hsiang Huang

Abstract—Previous studies in product-to-shelf assignment 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 data mining method for solving the product-to-shelf assignment problem with consideration of both product association rules and moving behavior of consumer. Specifically, the first task of this research is to develop a method to discover product association rules and consumers' moving behaviors of which are collected through RFID systems 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 items are reassigned to ensure customers can follow their preferred moving behaviors. Experimental result shows our proposed method can reassign minor items to suitable shelves and increase cross-selling opportunity of major and minor items.

Index Terms—Data mining, product-to-shelf assignment, moving and purchase patterns, customer behavior.

I. INTRODUCTION

To survive in competitive business, store managers should develop a retail mix strategy 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 [1]. Among that, product assortment and shelf space allocation are two important issues which dramatically affect customers' purchasing decisions [2]. In [3]-[6], the space elasticity has been widely used to construct the relationship between shelf space and product demand, and solved it with different optimization techniques. 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 variant types of 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 [7] 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. Cil [8] used buying association measure to create a category correlation matrix and applied the multidimensional scale technique to display the set of products in the store space. Although the association rules to assist managers in developing better layout for stores, their methods are 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 moving behaviors/patterns should be considered in solving product-to-shelf assignment problem. When shopping in a store, a customer moves 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.* [9] presented a multivariate clustering algorithm to explore the paths taken by individual shoppers in an actual grocery store, as provided by RFID (radio frequency identification) tags installed 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 moving 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 moving 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 [10]. 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 [11]. Specifically, the first task of this research is to develop a method to discover moving behavior of consumer, which includes both

Manuscript received August 20, 2014; revised June 24, 2015.

Chieh-Yuan Tsai is with the Department of Industrial Engineering and Management; Innovation Center for Big Data and Digital Convergence, Yuan Ze University, Chungli City, Taoyuan County, Taiwan, R.O.C. (e-mail: cytsai@yzu.edu.tw).

Sheng-Hsiang Huang is with the Department of Industrial Engineering and Management, Yuan Ze University, Chungli City, Taoyuan County, Taiwan, R.O.C. (e-mail: zevy6934@yahoo.com.tw).

high-utility mobile sequential patterns and product association rules, in the store. The second task 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 II describes some of the related works. Section III 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 IV, an empirical performance evaluation is conducted. Finally, conclusion is summarized in Section V.

II. RELATED WORK

A. Shelf Space Allocation

Shelf space is essential resource in logistic decisions and store management since high-quality space allocation can attract more consumers [5]. In the literatures, the space elasticity has been widely used to estimate the relationship between sales and allocated space. Space elasticity is the ratio of relative changes in unit sales to relative change in shelf space [12]. The measurement of space elasticity can be divided into two types which are direct elasticity (main effect) and cross-elasticity (cross-effect). Hansen and Heinsbroek [3] estimated the demand of products by space elasticity and constructed optimization models to select and allocate products. Corstjens and Doyle [4] extended this concept and developed a model to consider both space-elasticity and cross-elasticity. Yang and Chen [5] disregarded cross-elasticities and allowed the profit of each product to vary when allocated to different shelves by formulating the shelf-space allocation problem. Lim *et al.* [6] combined a local search technique with meta-heuristics to optimize the model of Yang and Chen [5]. However, using space elasticity for product assortment and shelf space allocation needs to estimate a great quantity of parameters which results in high cost and errors in the mathematical model.

B. Sequential Pattern Mining

Event-based data collected from scientific experiments or physical systems such as telecommunications networks, computer networks, and wireless sensor networks, and market basket transactions have characteristics of order. Agrawal and Srikant [13] first proposed the sequential pattern mining method to discover patterns that occur frequently in a sequence database. They provided an Apriori-like method using the candidate generation and pruning technique to mine sequential patterns. Then, many algorithms based on the Apriori method are proposed. Some researchers asserted different algorithms to improve the performance of sequential patterns mining because Apriori-based methods may generate too many candidates and need to scan database much times. FreeSpan [14] and PrefixSpan [15] are famous algorithms without generating any candidate pattern. Inspired by above works, some researchers start to modify those algorithms used in web environment and apply them to discover path traversal patterns in a radio-frequency identification (RFID) deployed supermarket/store. Using RFID is making it possible to track

customer behavior in stores, gather data, and gather information on the process of purchasing behavior. Larson *et al.* [9] and Liao and Lin [16] provided novel methods to capture shopping paths in a supermarket with RFID deployment.

In recent years, mining user behaviors in mobile environments is an emerging topic in the pattern mining field. Chen *et al.* [17] proposed a new data mining method that involves mining path traversal patterns in a distributed information-providing environment where documents or objects are linked together to facilitate interactive access. Tseng and Lin [18] proposed a novel data mining method, namely SMAP-Mine, that can discover patterns of sequential movement associated with requested services for mobile users in mobile web systems. However, the users' popular services may be different in different time periods. Lee *et al.* [19] proposed T-Map algorithm to find temporal mobile access patterns that can efficiently discover mobile user's temporal behavior patterns associated with location and requested services in different time intervals. Yun and Chen [10] proposed an algorithm TJ_{PF} to mine mobile sequential patterns, which takes both the moving patterns and purchase patterns of customers into consideration. Tseng *et al.* [20] proposed TMSP-Mine algorithm to efficiently generate the most proper time segmentation intervals based on the fitness function of genetic algorithm, and mine the mobile sequential patterns associated with moving paths and time intervals simultaneously in Location-Based Service (LBS) environments. On the other hand, the different customer groups may have mobile transaction sequences of different behaviors in the mobile environment. Lu and Tseng [21] proposed CMSP-Mine algorithm for efficiently discovering the Cluster-based Mobile Sequential Patterns (CMSPs) associated with moving paths and customer groups in LBS environments.

C. Utility Mining

Recently, utility mining is one of the most important research issues in data mining fields due to its ability to consider the nonbinary frequency values of items in transactions and different profit values for every item. Chan *et al.* [22] first proposed the concept of utility to capture highly desirable statistical patterns and level-wise itemset mining algorithm in association mining. Liu *et al.* [23] proposed a Two-Phase algorithm to efficiently prune down the number of candidates and can precisely obtain the complete set of high utility itemsets. Yao and Hamilton [24] proposed a utility based itemset mining approach which permits users to quantify their preferences concerning the usefulness of itemsets using utility values. Hu and Mojsilovic [25] proposed an algorithm for frequent itemset mining that identifies high-utility item combinations. Li *et al.* [26] proposed the Isolated Items Discarding Strategy (IIDS), which can be applied to existing level-wise utility mining method to reduce candidates and to improve performance. Some researches employed other frameworks such as tree-based framework to improve the performance of utility mining. Ahmed *et al.* [27] proposed a structure named IHUP-Tree which efficiently perform incremental and interactive high utility pattern mining. It avoids scanning

database multiple times and generating candidates in the mining process. Tseng *et al.* [28] proposed a novel algorithm named UP-Growth, for mining high utility itemsets with a set of techniques for pruning candidate itemsets. The information of high utility itemsets is maintained in a special data structure named UP-Tree such that the candidate itemsets can be generated efficiently with only two scans of the database. Shie *et al.* [29] proposed GUIDE algorithm which can find temporal maximal utility itemsets from data streams. They utilized TMUI-tree structure to capture the utility of each itemset with one-time scanning. Yen *et al.* (2011) [30] proposed a mechanism algorithm to reduce the mining space and can directly generate high utility itemsets from the tree structure without candidate generation. Although there are many researches about mobility pattern mining and utility mining, there is no research on the combination of the two topics. Shie *et al.* [11] first proposed the level-wise and tree-based algorithms which integrate sequential purchasing patterns with the moving paths and utility mining for finding high utility mobile sequential patterns in mobile environments.

III. THE PROPOSED METHOD

The framework of the proposed method consists of three main stages as shown in Fig. 1. The first stage 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 Fig. 2.

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 (task 1) 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.

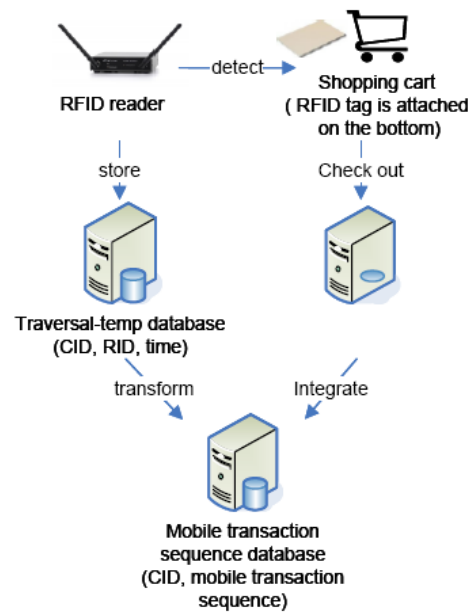


Fig. 2. Information architecture for collecting behaviour data.

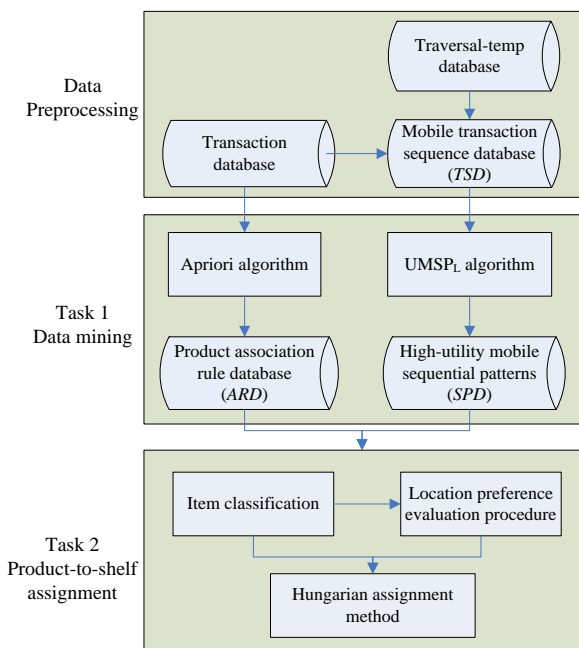


Fig. 1. The framework of the proposed product assignment method.

In the third stage (task 2), based on the product association rules and high-utility mobile sequential patterns generated, 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.

A. Store Layout

Fig. 3 provides a sample store layout for consideration. In the figure, a cabinet contains 6 shelves for product placement. RFID readers are deployed in appropriate positions to partition the aisles into several sections. The gray oval in Fig. 3 indicates the coverage area of an RFID reader. Let $I = \{i_1, i_2, \dots, i_g\}$ be the set of all product items sold in the store, $Z = \{z_1, z_2, \dots, z_y\}$ be the set of all shelves in the store, $S = \{S_1, S_2, \dots, S_n\}$ be the set of sections in the store, and $R = \{R_1, R_2, \dots, R_m\}$ be the set of RFID readers.

To collect behavior data of each customer, a customer is assigned a shopping cart attached with an activate RFID tag when he/she enters the store. When a cart passes through the coverage area of an RFID reader, the passing event with the form of (rid_i, t_i) will be recorded into the traversal-temp database where $rid_i \in R$ and t_i is the timestamp the cart is read. The system keeps recording the passing events until the customer completes shopping and checks out his/her purchased items at a counter. After the customer leaves the checkout counter, the purchase transaction of the customer will be stored into the transaction database with the form of $\{[it_1, q_1], [it_2, q_2], \dots, [it_h, q_h]\}$ where q_i is the purchased quantity of item $it_i \in I$.

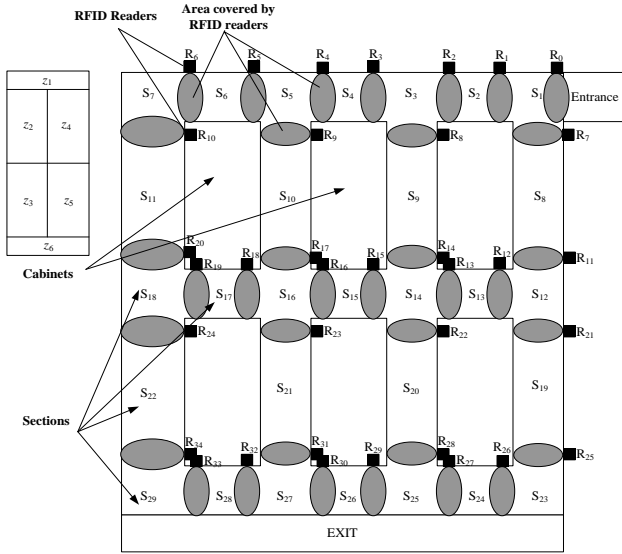


Fig. 3. The layout of an example store.

At the same time, the traversal path of the customer with the form of $\langle rid_1, rid_2, \dots, rid_p \rangle$ is retrieved from the traversal-temp database where $t_1 < t_2 < \dots < t_p$. Next, a traversal path represented by readers $\langle rid_1, rid_2, \dots, rid_p \rangle$ will be transformed into a traversal sequence represented by sections $\langle sid_1, sid_2, \dots, sid_p \rangle$ where $sid_p \in S$ according to the following transformation procedure. First, each reader in the traversal path is replaced by the two adjacent sections. The reason is that, when a reader detects a tag, the shopping cart might enter one of the two adjacent sections divided by the reader. After the passing events are replaced sequentially, the sections in two adjacent events representing where the customer comes from will be added into traversal sequence. Last, if a pair of repeated sections in a traversal sequence is found, the second one is deleted from the traversal sequence.

Table I shows five traversal paths collected in the store layout of Fig. 3. For instance, the traversal path with customer id 1 indicates the customer with a shopping cart passes through entrance R_0 (1st event), reader R_1 (2nd event), reader R_2 (3rd event), reader R_3 (4th event) and so on. To transform the traversal path to a traversal sequence, the 1st and 2nd events $\langle R_0, R_1 \rangle$ is replaced as $\langle (\text{entrance}, S_1), (S_1, S_2) \rangle$ since the adjacent sections of R_0 is $\{\text{entrance}, S_1\}$ and the adjacent sections of R_1 is $\{S_1, S_2\}$. Repeated sections S_1 is found among $\{\text{entrance}, S_1, S_1, S_2\}$. Then, $\langle (\text{entrance}, S_1), (S_1, S_2) \rangle$ is transformed as $\langle \text{entrance}, S_1, S_2 \rangle$. Events 4 to 6 in the traversal path $\langle R_3, R_3, R_8 \rangle$ show another situation. It is clear

that the cart passes R_3 twice. Thus, events 4 to 6 will be transformed into $\langle (S_3, S_4), (S_4, S_3), (S_3, S_9) \rangle$. Since two pairs of sections S_4 and S_3 are found, the second one will be deleted. Therefore, it will be transferred as $\langle S_3, S_4, S_3, S_9 \rangle$. Table II shows the traversal sequence transformed from the traversal path in Table I after conducting the proposed transformation procedure.

TABLE I: THE EXAMPLE TRAVERSAL PATHS IN THE TRAVERSAL-TEMP DATABASE

CID	Traversal path
1	$\langle R_0, R_1, R_2, R_3, R_3, R_8, R_14, R_{22}, R_{28} \rangle$
2	$\langle R_0, R_7, R_{11}, R_{12}, R_{13}, R_{15}, R_{16}, R_{23}, R_{31} \rangle$
3	$\langle R_0, R_7, R_{11}, R_{12}, R_{13}, R_{15}, R_{16}, R_{18}, R_{18}, R_{23}, R_{31} \rangle$
4	$\langle R_0, R_1, R_2, R_3, R_4, R_9, R_{17}, R_{16}, R_{15}, R_{22}, R_{28} \rangle$
5	$\langle R_0, R_7, R_{11}, R_{12}, R_{13}, R_{15}, R_{16}, R_{23}, R_{31} \rangle$

TABLE II: THE EXAMPLE TRAVERSAL SEQUENCES TRANSFORMED FROM TRAVERSAL PATHS IN TABLE I

CID	Traversal sequence
1	$\langle S_1, S_2, S_3, S_4, S_3, S_9, S_{14}, S_{20}, S_{25} \rangle$
2	$\langle S_1, S_8, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}, S_{21}, S_{27} \rangle$
3	$\langle S_1, S_8, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}, S_{17}, S_{16}, S_{21}, S_{27} \rangle$
4	$\langle S_1, S_2, S_3, S_4, S_5, S_{10}, S_{16}, S_{15}, S_{14}, S_{20}, S_{25} \rangle$
5	$\langle S_1, S_8, S_{12}, S_{13}, S_{14}, S_{15}, S_{16}, S_{21}, S_{27} \rangle$

With the purchasing transaction and traversal sequence, a mobile transaction sequence of a customer can be derived and defined as $TS = \langle T_1, T_2, \dots, T_n \rangle$ where transaction $T_j = (Se_j; \{[it_{j1}, q_{j1}], [it_{j2}, q_{j2}], \dots, [it_{jh}, q_{jh}]\})$ represents that a customer purchases $\{[it_{j1}, q_{j1}], [it_{j2}, q_{j2}], \dots, [it_{jh}, q_{jh}]\}$ in section Se_j where q_{jp} is the purchased quantity of p th item $it_{jp} \in I$ in transaction T_j . A path is denoted as $Se_1 Se_2 \dots Se_r$, where $Se_j \in S$ and $1 \leq j \leq r$. All the mobile transaction sequence (TS) will be stored in the mobile transaction sequence database (TSD).

Table III shows five mobile transaction sequence (TSS) in the mobile transaction sequence database (TSD). For example, customer CID 1 bought one item i_5 in section S_2 , five items i_{11} in section S_4 , and six items i_{58} in section S_{20} , customer CID 2 bought five items i_{22} and three items i_{24} in section S_8 , five items i_{38} in section S_{13} , and so on.

TABLE III: THE MOBILE TRANSACTION SEQUENCES IN THE TSD

CID	Mobile transaction sequence
1	$\langle (S_1; \emptyset), (S_2; \{[i_5, 1]\}), (S_3; \emptyset), (S_4; \{[i_{11}, 5]\}), (S_5; \emptyset), (S_9; \emptyset), (S_{14}; \emptyset), (S_{20}; \{[i_{58}, 6]\}), (S_{25}; \emptyset) \rangle$
2	$\langle (S_1; \emptyset), (S_8; \{[i_{22}, 5], [i_{24}, 3]\}), (S_{12}; \emptyset), (S_{13}; \{[i_{38}, 5]\}), (S_{14}; \emptyset), (S_{15}; \{[i_{43}, 2]\}), (S_{16}; \emptyset), (S_{21}; \{[i_{62}, 3]\}), (S_{27}; \emptyset) \rangle$
3	$\langle (S_1; \emptyset), (S_8; \{[i_{22}, 3]\}), (S_{12}; \emptyset), (S_{13}; \{[i_{38}, 8]\}), (S_{14}; \emptyset), (S_{15}; \{[i_{43}, 3]\}), (S_{16}; \emptyset), (S_{17}; \{[i_{50}, 3]\}), (S_{16}; \emptyset), (S_{21}; \{[i_{62}, 2]\}), (S_{27}; \emptyset) \rangle$
4	$\langle (S_1; \emptyset), (S_2; \{[i_5, 1]\}), (S_3; \emptyset), (S_4; \{[i_{11}, 3]\}), (S_5; \emptyset), (S_{10}; \emptyset), (S_{16}; \emptyset), (S_{15}; \{[i_{43}, 3]\}), (S_{14}; \emptyset), (S_{20}; \{[i_{58}, 3]\}), (S_{25}; \emptyset) \rangle$
5	$\langle (S_1; \emptyset), (S_8; \{[i_{22}, 3], [i_{24}, 2]\}), (S_{12}; \emptyset), (S_{13}; \{[i_{38}, 6]\}), (S_{14}; \emptyset), (S_{15}; \{[i_{43}, 2]\}), (S_{16}; \emptyset), (S_{21}; \{[i_{62}, 3]\}), (S_{27}; \emptyset) \rangle$

B. Association Rules Generation

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 [31], 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 single 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).

C. High-Utility Mobile Sequential Patterns Generation

The $UMSP_L$ (high Utility Mobile Sequential Pattern by a Level-wised method) algorithm proposed by Shie *et al.* [11] 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 [23], while the fourth 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 Fig. 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).

TABLE IV: 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 V: TRANSFORMED MOBILE TRANSACTION SEQUENCE DATABASE TD

CID	Sequence of $WULIs$	Path	SU
1'	$\langle S_2; t_1; 1 \rangle, \langle S_4; t_2; 3 \rangle, \langle S_{20}; t_8; 7 \rangle$	$S_2S_3S_4S_5S_9S_{14}S_{20}$	93
2'	$\langle S_8; t_3; t_4; t_{10}; 1 \rangle, \langle S_{13}; t_5; 3 \rangle, \langle S_{15}; t_6; 5 \rangle, \langle S_{21}; t_9; 7 \rangle$	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{21}$	153
3'	$\langle S_8; t_3; 1 \rangle, \langle S_{13}; t_5; 3 \rangle, \langle S_{15}; t_6; 5 \rangle, \langle S_{17}; t_7; 7 \rangle, \langle S_{21}; t_9; 9 \rangle$	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{17}S_{16}S_{21}$	160
4'	$\langle S_2; t_1; 1 \rangle, \langle S_4; t_2; 3 \rangle, \langle S_{15}; t_6; 7 \rangle, \langle S_{20}; t_8; 9 \rangle$	$S_2S_3S_4S_5S_{10}S_{16}S_{15}S_{14}S_{20}$	89
5'	$\langle S_8; t_3; t_4; t_{10}; 1 \rangle, \langle S_{13}; t_5; 3 \rangle, \langle S_{15}; t_6; 5 \rangle, \langle S_{21}; t_9; 7 \rangle$	$S_8S_{12}S_{13}S_{14}S_{15}S_{16}S_{21}$	134

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 IV and Table V, respectively. In TD , the original mobile transaction sequences are parsed into the sequences of section-itemsets and paths. For instance, $\langle S_4; t_2; 3 \rangle$ 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.

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 $SWU(X_T) \geq \epsilon$

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 $SWU(X_T) \geq \epsilon$

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

Fig. 4. Third step of the $UMSP_L$ algorithm.

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 $\langle S_{21}; t_9 \rangle$ are respectively shown in Fig. 5(a) and 5(b). Fig. 5(a) indicates five 2- $WUMSPs$ marked with solid lines are generated, while Fig. 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 $\langle S_{21}; t_9 \rangle$ are found.

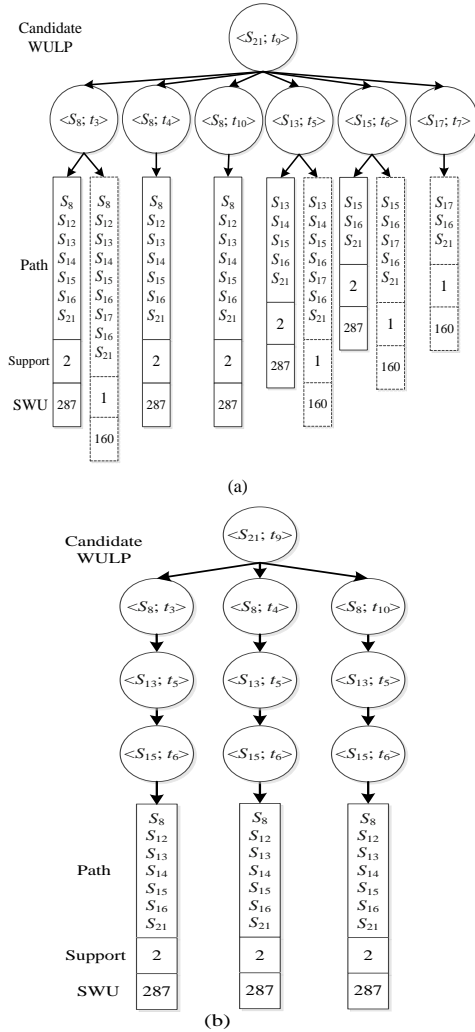


Fig. 5. (a) 2-candidate tree of $\langle S_{21}; t_9 \rangle$; (b) 4-candidate tree of $\langle S_{21}; t_9 \rangle$.

D. 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 I: 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 II: 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 .

To explain how to generate the set of major items MA and the set of minor items MI , the high-utility mobile sequential patterns in Table VI and the product association rules in Table VII are used as an example. In Table VI, there are 3 high-utility mobile sequential patterns which are $\langle \{ \langle S_8; i_{22} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$, $\langle \{ \langle S_8; i_{22} \rangle \langle S_{13}; i_{38} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$ and $\langle \{ \langle S_8; i_{22} \rangle \langle S_{13}; i_{38} \rangle \langle S_{15}; i_{43} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$. According to the definition of a major item, $MA = \{ i_{22}, i_{38}, i_{43}, i_{62} \}$ can be identified. Next, based on the definition of a minor item, items $i_8, i_{25}, i_{38}, i_{43}, i_{52}, i_{62}, i_{70}$ are candidate minor items since the items on the consequent of all association rules belong to major items. However, items i_{38}, i_{43} , and i_{62} are major items already. Therefore, the set of minor items is $MI = \{ i_8, i_{25}, i_{52}, i_{70} \}$.

TABLE VI: THE SAMPLE HIGH-UTILITY MOBILE SEQUENTIAL PATTERNS

Pattern ID	High-utility mobile sequential pattern (UMSP)
1	$\langle \{ \langle S_8; i_{22} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$
2	$\langle \{ \langle S_8; i_{22} \rangle \langle S_{13}; i_{38} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$
3	$\langle \{ \langle S_8; i_{22} \rangle \langle S_{13}; i_{38} \rangle \langle S_{15}; i_{43} \rangle \langle S_{21}; i_{62} \rangle \}; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$

TABLE VII: THE EXAMPLE PRODUCT ASSOCIATION RULES

Rule ID	Association rule	Rule ID	Association rule
1	$i_{22} \rightarrow i_{25}$	6	$i_{43} \rightarrow i_{52}$
2	$i_{22} \rightarrow i_{38}$	7	$i_{43} \rightarrow i_{62}$
3	$i_{38} \rightarrow i_8$	8	$i_{62} \rightarrow i_{38}$
4	$i_{38} \rightarrow i_{25}$	9	$i_{62} \rightarrow i_{52}$
5	$i_{38} \rightarrow i_{43}$	10	$i_{62} \rightarrow i_{70}$

E. Product Reassignment

According to definitions I and II, 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.

Take high-utility mobile sequential pattern $UMSP_2 = \langle \{ \langle S_8; i_{22} \rangle \langle S_{13}; i_{38} \rangle \langle S_{21}; i_{62} \rangle \} ; S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21} \rangle$ in Table VI as an example. Assume there are eight sections of S_8 , S_{12} , S_{13} , S_{14} , S_{15} , S_{16} , S_{21} , and S_{25} in the store. According to association rule 3 in Table VII, we know that minor item i_8 is related to major item i_{38} . Since minor item i_8 is already located in section S_{13} , according to $UMSP_2$ it does not need to be relocated. That is, movement distance $D_{i_{38},UMSP_2}^{i_8,S_{13}} = 0$. If minor item i_8 is placed in section S_{12} or S_{14} , $D_{i_{38},UMSP_2}^{i_8,S_{12}} = D_{i_{38},UMSP_2}^{i_8,S_{14}} = 1$ since the movement distance from S_{13} to section S_{12} or S_{14} is 1 according to the path information ($S_8 S_{12} S_{13} S_{14} S_{15} S_{16} S_{21}$) in $UMSP_2$. Similarly, $D_{i_{38},UMSP_2}^{i_8,S_8} = D_{i_{38},UMSP_2}^{i_8,S_{15}} = 2$, $D_{i_{38},UMSP_2}^{i_8,S_{16}} = 3$, and $D_{i_{38},UMSP_2}^{i_8,S_{21}} = 4$. Finally, $D_{i_{38},UMSP_2}^{i_8,S_{25}} = \beta$ since section S_{25} does not appear in $UMSP_2$.

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:

$$W_{k,m}^{j,n} = \frac{D_{k,m}^{j,n}}{\beta}, \quad 0 \leq D_{k,m}^{j,n} \leq \beta \quad (1)$$

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

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 [32] is adopted in this study. The Hungarian method is a combinatorial optimization algorithm that can solve the assignment problem.

IV. IMPLEMENTATION AND EXPERIMENTAL RESULTS

A. Data Description

A simplified supermarket as illustrated in Fig. 6 is used to demonstrate the feasibility of the proposed product-to-shelf 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 III.A. 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.

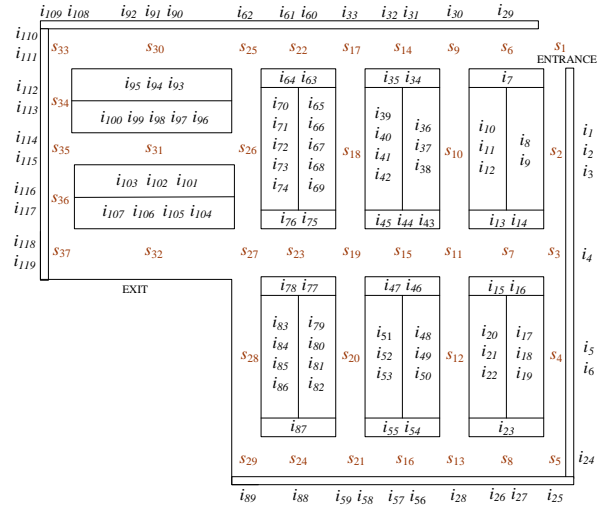


Fig. 6. Physical location of all products.

B. Experimental Results

Based on the transactions from the generator, 21 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 VIII. Next, $UMSP_L$ algorithm is applied to generate high-utility mobile sequential patterns based on the utility data in Table IX. There are 14 high-utility mobile sequential patterns generated when minimum support count is 6 and the minimum utility is 300. Part of the high-utility mobile sequential patterns is shown in Table X.

TABLE VIII: ASSOCIATION RULE (10%, 60%)

ID	Rule	ID	Rule	ID	Rule
1	$i_4 \rightarrow i_{43}$	5	$i_{15} \rightarrow i_{39}$	9	$i_{62} \rightarrow i_{93}$
2	$i_{43} \rightarrow i_4$	6	$i_{39} \rightarrow i_{15}$	10	$i_{93} \rightarrow i_{62}$
3	$i_4 \rightarrow i_{15}$	7	$i_{51} \rightarrow i_{66}$
4	$i_{15} \rightarrow i_4$	8	$i_{66} \rightarrow i_{51}$	21	$i_{106} \rightarrow i_{86}$

TABLE IX: THE UTILITY TABLE FOR ITEMS

Item	Profit	Item	Profit	Item	Profit
i_1	100	i_5	200	i_9	30
i_2	200	i_6	800	i_{10}	80
i_3	150	i_7	30
i_4	120	i_8	20	i_{119}	20

TABLE X: HIGH-UTILITY MOBILE SEQUENTIAL PATTERN (MINIMUM SUPPORT COUNT=6)

PID	Pattern
1	$\langle \{ \langle S_3; i_4 \rangle \langle S_7; i_{15} \rangle \langle S_{12}; i_{50} \rangle \langle S_{21}; i_{58} \rangle \} ; S_1, S_2, S_3, S_7, S_{11}, S_{12}, S_{13}, S_{21} \rangle$
2	$\langle \{ \langle S_3; i_4 \rangle \langle S_7; i_{15} \rangle \langle S_{12}; i_{50} \rangle \} ; S_1, S_2, S_3, S_7, S_{11}, S_{12} \rangle$
3	$\langle \{ \langle S_3; i_4 \rangle \langle S_7; i_{15} \rangle \langle S_{21}; i_{58} \rangle \} ; S_1, S_2, S_3, S_7, S_{11}, S_{21} \rangle$
...	...
13	$\langle \{ \langle S_{25}; i_{62} \rangle \langle S_{32}; i_{106} \rangle \} ; S_1, S_6, S_9, S_{14}, S_{17}, S_{22}, S_{25}, S_{32} \rangle$
14	$\langle \{ \langle S_{18}; i_{66} \rangle \langle S_{32}; i_{106} \rangle \} ; S_1, S_6, S_9, S_{14}, S_{17}, S_{18}, S_{19}, S_{27}, S_{32} \rangle$

After association rules and high-utility mobile sequential patterns are generated, major items and minor items can be found. In this simulation, 8 major items including $i_4, i_{15}, i_{50}, i_{58}, i_{62}, i_{66}, i_{86}$ and i_{106} are found. In addition, 6 minor items including $i_{17}, i_{39}, i_{43}, i_{51}, i_{87}$ and i_{93} are identified. The major items should not be rearranged from their original positions because doing so would disrupt frequent customer visiting behaviors (i.e. high-utility mobile sequential patterns). Therefore, only minor items will be rearranged. With major and minor items, the threshold of maximum section movement β is set as 3, the location preference weight matrix $[f_{i,y}]$ can be obtained according to Equation (1).

The final stage of the proposed product-to-shelf allocation method is to reassign minor items to their best shelf location using Hungarian method. Table XI shows the reassignment result after taking Hungarian method. We find that i_{17} is strongly related to major item i_{58} , i_{39} is strongly related to major item i_{15} , i_{43} is strongly related to major item i_4 , and i_{51} is strongly related to major item i_{66} . Thus, the four minor items ($i_{17}, i_{39}, i_{43}, i_{51}$) are re-organized to the locations close to their associated major items ($i_{58}, i_{15}, i_4, i_{66}$). Minor item i_{93} does not change shelf location since i_{93} already located on the shelf very close to its major item i_{62} at the original layout. Minor item i_{87} is not assigned to the best shelf z_{43} since the location preference weights are calculated based on average concept. The minor items are rearranged to be in closer proximity to their corresponding major items, thus increasing the likelihood of successful cross-selling.

TABLE XI: RESULT OF ASSIGNMENT

Minor Item	Original Shelf	New Shelf
i_{17}	z_{12}	z_{38}
i_{39}	z_{21}	z_{22}
i_{43}	z_{22}	z_{12}
i_{51}	z_{25}	z_{21}
i_{87}	z_{38}	z_{25}
i_{93}	z_{43}	z_{43}

V. CONCLUSION

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.

Some potential extensions for this research are as follows. First, in the current study, all products on a given shelf are limited to the same major category (e.g., "Beverage"). Future work could consider grouping items from multiple product classes. Second, this study assumes product volumes are identical and can be smoothly exchanged. However, in

practice, different products might be displayed in different volumes, and future work can consider this issue. Third, this study solves the product-to-shelf assignment using the Hungarian method, which would run too slowly given large quantities of data, and future work can test different assignment algorithms.

REFERENCES

- [1] M. Levy and B. A. Weitz, *Retailing Management*, Chicago: Irwin, 1995.
- [2] M. H. Yang, "An efficient algorithm to allocate shelf space," *European Journal of Operational Research*, vol. 131, no. 1, pp. 107-118, 1999.
- [3] P. Hansen and H. Heinsbroek, "Product selection and space allocation in supermarket," *European Journal of Operational Research*, vol. 3, no. 6, pp. 474-484, 1979.
- [4] M. Corstjens and P. Doyle, "A model for optimizing retail space allocations," *Management Science*, vol. 27, no. 7, pp. 822-833, 1981.
- [5] M. H. Yang and W. C. Chen, "A study on shelf space allocation and management," *International Journal of Production Economics*, pp. 309-317, 1999.
- [6] A. Lim, B. Rodrigues, and X. Zhang, "Metaheuristics with local search technique for retail shelf-space optimization," *Management Science*, vol. 50, no. 1, pp. 117-131, 2004.
- [7] M. C. Chen and C. P. Lin, "A data mining approach to product assortment and shelf space allocation," *Expert Systems with Applications*, vol. 32, no. 4, pp. 976-986, 2007.
- [8] I. Cil, "Consumption universes based supermarket layout through association rule mining and multidimensional scaling," *Expert Systems with Applications*, vol. 39, no. 10, pp. 8611-8625, 2012.
- [9] J. S. Larson, E. T. Bradlow, and P. S. Fader, "An exploratory look at supermarket shopping paths," *International Journal of Research in Marketing*, vol. 22, no. 4, pp. 395-414, 2005.
- [10] C. H. Yun and M. S. Chen, "Mining mobile sequential patterns in a mobile commerce environment," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 37, no. 2, pp. 278-295, 2007.
- [11] B. E. Shie, H. F. Hsiao, and V. S. Tseng, "Efficient algorithms for discovering high utility user behavior patterns in mobile commerce environments," *Knowledge Information Systems*, pp. 1-25, 2012.
- [12] R. C. Curhan, "The relationship between shelf space and unit sales in supermarkets," *Journal of Marketing Research IX*, pp. 406-412, 1972.
- [13] R. Agrawal and R. Srikant, "Mining sequential patterns," *Proceedings of ICDE*, pp. 360-371, 1995.
- [14] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," in *Proc. the 2000 ACM-SIGMOD International Conference on Management of Data*, Dallas, TX., 2000.
- [15] J. Pei, J. Han, H. Pinto, Q. Chen, U. Dayal, and M. C. Hsu, "PrefixSpan: Mining sequential patterns efficiently by prefix-projected pattern growth," in *Proc. the 2001 International Conference on Data Engineering*, pp. 215-224, 2001.
- [16] I. E. Liao, and W. C. Lin, "Shopping path analysis and transaction mining based on RFID technology," *RFID Eurasia*, Istanbul, Turkey, pp. 292-296, 2007.
- [17] M. S. Chen, J. S. Park, and P. S. Yu, "Efficient data mining for path traversal patterns," *IEEE Transactions on Knowledge and Data Engineering*, vol. 10, no. 2, pp. 209-221, 1998.
- [18] V. S. M. Tseng and K. W. C. Lin, "Mining sequential mobile access patterns efficiently in mobile web systems," in *Proc. International Conference on Advanced Information Networking and Application*, vol. 54, no. 2, pp. 762-767, 2005.
- [19] S. C. Lee, J. Paik, J. Ok, I. Song, and U. M. Kim, "Efficient mining of user behaviors by temporal mobile access patterns," *IJCSNS International Journal of Computer Science and Network Security*, vol. 7, no. 2, pp. 285-291, 2007.
- [20] V. S. Tseng, E. H. C. Lu, and C. H. Huang, "Mining temporal mobile sequential patterns in location-based service environments," in *Proc. the International Conference on Parallel and Distributed Systems*, 2007, pp. 1-8.
- [21] E. H. C. Lu and V. S. Tseng, "Mining cluster-based mobile sequential patterns in location-based service environments," in *Proc. the IEEE International Conference on Mobile Data Management*, 2009, pp. 273-278.
- [22] R. Chan, Q. Yang, and Y. D. Shen, "Mining high utility itemsets," in *Proc. IEEE International Conference on Data Mining*, 2003, pp. 19-26.

- [23] Y. Liu, W. K. Liao, and A. Choudhary, "A fast high utility itemsets mining algorithm," in *Proc. the 1st International Workshop on Utility-Based Data Mining*, 2005, pp. 90-99.
- [24] H. Yao and H. J. Hamilton, "Mining itemset utilities from transaction databases," *Data and Knowledge Engineering*, pp. 603-626, 2006.
- [25] J. Hu and A. Mojsilovic, "High-utility pattern mining: A method for discovery of high-utility item sets," *Pattern Recognition*, vol. 40, no. 11, pp. 3317-3324, 2007.
- [26] Y. C. Li, J. S. Yeh, and C. C. Chang, "Isolated items discarding strategy for discovering high utility itemsets," *Data and Knowledge Engineering*, vol. 64, no. 1, pp. 198-217, 2008.
- [27] C. F. Ahmed, S. K. Tanbeer, B. S. Jeong, and Y. K. Lee, "Efficient tree structures for high utility pattern mining in incremental databases," *IEEE Transactions on knowledge and Data Engineering*, vol. 21, no. 12, pp. 1708-1721, 2009.
- [28] V. S. Tseng, C. W. Wu, B. E. Shie, and P. S. Yu, "UP-growth: An efficient algorithm for high utility itemsets mining," in *Proc. the 16th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, 2010, pp. 253-262.
- [29] B. E. Shie, V. S. Tseng, and P. S. Yu, "Online mining of temporal maximal utility itemsets from data streams," in *Proc. the 2010 ACM Symposium on Applied Computing*, pp. 1622-1626, 2010.
- [30] S. J. Yen, C. C. Chen, and Y. S. Lee, "A fast algorithm for mining high utility itemsets," in *Proc. the 2011 Workshop on Behavior Informatics*, 2011.
- [31] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. the 20th International Conference on Very Large Data Bases (VLDB)*, Santiago, Chile, 1994, pp. 487-499.
- [32] H. W. Kuhn, "The hungarian method for the assignment problem," *Naval Research Logistics Quarterly*, vol. 2, pp. 83-97, 1955.



applications, and product data management.

Chieh-Yuan Tsai is a professor in the Department of Industrial Engineering and Management at Yuan-Ze University, Taiwan. He received his M.S. and Ph.D. degrees in the Department of Industrial and Manufacturing Systems Engineering from the University of Missouri-Columbia, USA. His research activities include data mining, customer relationship management (CRM), RFID technology and



Sheng-Hsiang Huang is a Ph.D. candidate in the Department of Industrial Engineering and Management at Yuan Ze University, Taiwan. He received his master degree at the same institute. His research activities include data mining, shelf-space allocation, and inventory management.