Supermarket Product Placement Strategies Based on Association Rules

Francisco Javier Moreno Arboleda, Georgia Garani, Andrés Felipe Arboleda Correa

***Abstract*— The way products are placed in a supermarket can be effective in increasing sales and profit. A reasonable approach is to group together items that are likely to be purchased together. Thus, managers with the support of mining methods and techniques, can assist customers locating the products they want to buy in an easy and quick way. Many product placement strategies have been proposed over the years to leverage an effective and efficient way to achieve this goal. In this paper, association rules for product arrangement in supermarkets are studied and an algorithm based on such rules is proposed. The algorithm considers several factors, such as the number of units sold of each product, a hierarchical structure for product classification developed by the United Nations Department of Economic and Social Affairs, a set of association rules generated from sales, and a set of constraints that restrict some products to be placed physically close to each other in a supermarket, even if they are usually purchase together. Real public sales data of a supermarket were used for the experiments, where the proposed algorithm is applied for the generation of supermarket layouts. The results show that some supermarket departments may share the same products or product categories.**

***Index Terms*— Association rules; data mining algorithms; Apriori algorithm; market basket analysis.**

I. Introduction

THE layout of a supermarket can influence the purchasing behavior of its customers, its operational efficiency [1], and its profit [2], which is one of its main objectives (profit maximization [3]). A supermarket is usually redesigned to meet the changing needs of the market, e.g., entry of new brands and products, changes in customers' shopping preferences, marketing policies (promotions, events on special days), physical changes of the store (expansions or reductions), among others. However, the layout of a supermarket is a challenging task [4]. For example, the products (or product categories) for sale and their arrangement in a supermarket must be considered.

In [5] some works that focus on the layout of supermarkets are analyzed [2], [4], [6]–[9]. Two approaches are considered: i) the classical one, where supermarkets are traditionally organized in departments, i.e., in a department, products that share some functional characteristics are grouped together, e.g., beverages, dairy, meat, bakery, and fruits and ii) the one proposed by Cil [2], where products are grouped around *consumption universes*. Thus, instead of finding coffee in the beverages department, cheese in the dairy department, ham in the meat department, bread in the bakery department, and orange in the fruits department, one could find all these products in the "Breakfast Universe". Other examples of consumption universes are the "Baby Universe" and the "Countryside Universe".

On the other hand, for the layout of a supermarket, association rules (ARs) [10] can be used to determine which products or product categories should be located close together (e.g., in the same supermarket department). For this purpose, a sales transaction database is examined, and it is determined which products or product categories are most frequently sold together [2], [11].

The Apriori algorithm is one of the most widely used to obtain ARs. The algorithm obtains the frequent sets of items (in our case products), from their occurrence in the transactions (sales) but does not consider the profit generated by the sale of a product nor the number of units sold of each product (NUSP). These two aspects are of interest to the analyst because it may happen that a product: i) appears only in a few sales, ii) is sold in large volumes, and iii) is determinant for the profit of the supermarket.

In this paper, we propose an algorithm based on ARs for the generation of several layouts for a supermarket. Our algorithm considers: i) the NUSP for the generation of the ARs, ii) a hierarchical structure for the classification of the products developed by the United Nations Department of Economic and Social Affairs (UN DESA) [12], iii) the selection of interesting ARs for the analyst (that meet certain thresholds), and iv) a set of constraints specified by the analyst, which establish that certain products or product categories *should not* be located in the same department. Unlike other proposals (see section 2), we consider all these four aspects. In addition, in our proposal the analysts can generate different layouts depending on the number of supermarket departments and can choose the product level classification for creating the supermarket layouts, e.g., create the layouts with specific products (e.g., ACME TV S345) or with product categories (e.g., TVs or electronics), i.e., with a higher level of classification.

The paper is organized as follows: in section 2 we present related works, in section 3 we show an example and present some definitions of the ARs, in section 4 we propose our algorithm and apply it to our example, in section 5 we present experiments with the generation of layouts where we apply our algorithm on a real public data set of sales of a supermarket. Finally, in section 6 we conclude and propose future work.

Manuscript received November 15, 2023; revised April 11, 20024.

Francisco Javier Moreno Arboleda is an associate professor at the Departamento de Ciencias de la Computación y de la Decisión, Universidad Nacional de Colombia, Sede Medellín, Colombia (phone: 604-425-5376; e-mail: fjmoreno@unal.edu.co).

Georgia Garani is an associate professor at the Digital Systems Department of the University of Thessaly, Greece. (garani@uth.gr).

Andrés Felipe Arboleda Correa is a master's graduate in computer science from Universidad Nacional de Colombia, Sede Medellín, Colombia. (afarboledac@unal.edu.co).

II. Related Work

In Table I we compiled works focused on: i) ARs where the NUSP is considered for their generation and ii) the application of ARs for the layout of facilities, including supermarkets. In a previous work [5] we presented a detailed analysis of these works.

III. Example

Suppose we want to redesign the departments of a supermarket and we are going to consider its sales history. For this example, 24 sales are considered. Table II shows some of them. The complete table can be found in *https://drive.google.com/file/d/1E4pezd3n4CW4MVc1NpxI8wWm6oT1GJ-O*.

TABLE II
Sales sample

Id	Sale
1	{1 Skim milk, 5 Red gala apples, 5 Granny smith apples, 1 Saltine cookies, 1 Laundry soap}
2	{1 UHT milk, 1 Wafer cookies}
3	{1 UHT milk, 8 Red gala apples, 1 Saltine cookies, 1 Body soap}
…	…
24	{1 Skim milk, 3 Brooms}

The products from the sales of Table II can be classified into a product category tree: sections, divisions, groups, classes, and subclasses [12]. In Fig. 1 (redrawing based on [12]) we show the categorization at the subclass level for "Granny smith apple" and "Red gala apple". In Table III we show statistics for the products of Table II after classifying them into the corresponding subclasses.

TABLE I
Comparison of analyzed works

Ref.	Approach	Proposal	Remarks
[13]	ARs induced by NUSP	A framework that generates ARs using the Apriori algorithm	It considers the NUSP to generate the ARs
[14]	Sales trends and quantity forecast	A data mining system to determine sales trends and quantity forecast using ARs using the Apriori algorithm	It can help to discover the trends (growth, stability, decline) of product sales. It considers the NUSP to generate the ARs
[15]	Algorithm for finding frequent itemsets	A Q-TID (Quantity-Transaction IDentifiers) tree algorithm for finding frequent itemsets	The proposed algorithm required 70% less execution time than the Apriori algorithm and 60% less than the FP-Growth (Frequent Pattern Growth) algorithm. It considers the NUSP to generate the ARs
[16]	AR mining framework using item weight	A framework, QBARM (Quantity Based Association Rule Mining) that generates ARs	It considers the NUSP and the item weight (profit) to generate the ARs
[17]	AR mining based on profit and NUSP	An algorithm that generates ARs	It considers the product profit and the NUSP to generate the ARs
[18]	Facility layout	They applied MBA (Market Basket Analysis) to design a facility for an amusement arcade	Two layouts for an amusement arcade are proposed considering the frequency with which the games are played and their game category
[2]	Supermarket layout	A supermarket layout based on the ARs among product categories and MDS (Multidimensional Scaling) [4]	Only one layout is proposed
[19]	Multiple store environment	A method that generates ARs in a multi-store environment	The ARs include information about the store (location) and the time when the ARs held. No layout is proposed
[20]	MBA for supermarket products	The Apriori and K-Apriori algorithms are compared based on the frequent itemsets and ARs generated	It presents a case study to evaluate the Apriori and K-Apriori algorithms. The MBA using K-Apriori for Anantha stores (located in India) improved its overall revenue. Although it is not indicated how much improvement there was
[21]	AR mining and a customer-oriented approach	The aim is to show that AR mining can be applied in a customer-oriented approach. The ARs were determined among product categories	Proactive customer-oriented practices are obtained to optimize the sales and the customer satisfaction in a shop. No layout is proposed
[6]	Store/supermarket layout	Layout of a store/supermarket into departments where the products that share some functional characteristics or the same origin are grouped together	The store layout is static since it does not consider the evolution of sales. ARs are not used
[4]	Store/supermarket layout	ARs between product categories are generated and the MDS is applied to propose the layout of a supermarket	This approach is based on the principle that the conjoint use of products will produce conjoint buying. Only one layout is proposed
[8]	Store/supermarket layout	An algorithm (HUIM, High-Utility Itemset Mining) to rearrange a store layout and to determine the relationships among product categories	It considers the profit, and the NUSP. Only one layout is proposed
[9]	Facility layout	A weighted AR-based data mining approach for a facility layout	It considers the demand and the efficiency of material handling equipment as weighting criteria. Their approach can be applied to multi-objective facility layout problems, e.g., minimization of the total material handling costs and minimization of the time [22]. Only one layout is proposed

0. Agriculture, forestry and fishery products → Section
    01. Products of agriculture, horticulture and market gardening → Division
        013. Fruits and nuts → Group
            01341. Apples → Class
                - Granny smith apple → Subclass
                - Red gala apple
...
2. Food products, beverages and tobacco; textiles, apparel and leather products
    22. Dairy products and egg products
    ...

Fig. 1. Product category tree (redrawing based on [12]).

In this example we assumed that the subclass Processed liquid milk is purchased in all 24 sales. According to the data in Table III we observed that Processed liquid milk is usually purchased together with the subclass Biscuits in 70.8% of the sales, with the subclass Soap in 33% of the sales, and with the subclass Apples in 25% of the sales. This suggests that Processed liquid milk could be in two or more departments of the supermarket, e.g., in the Bakery department and in the Fruit department (but not in the Cleaning & Household department, for sanitary reasons). This example shows that the sets of products that are frequently purchased together is an insufficient aspect to solve the layout problem of a supermarket, i.e., which products should be placed close to each other? [3], since there are rules, e.g., chemicals are not allowed to be placed next to food, or traditions, e.g., men's and women's underwear are not placed in the same department. If these rules or traditions are violated, this may cause damage to health (fruits next to detergents) or look strange (e.g., men's underpants next to women's underwear). Accordingly, it is appropriate to specify a set of products (or product categories) constraints to specify which products *should not* be close to each other, such as dairy products and detergents, even though they frequently appear together in sales.

Next, we present some essential concepts of ARs. In the context of a supermarket, ARs can be generated from its sales and indicate actual consumer behavior [2]. Each of the products (or product categories) that are part of the sales is called an *item* and a set of these is called an *itemset.*

An AR has two parts: *left-hand side* (LHS) and *right-hand side* (RHS) and is represented as follows: $A \rightarrow B$, where A and B are itemsets, A is the LHS and B the RHS. For example, {*Processed liquid milk*} → {*Apples*}.

The problem of identifying ARs is divided into finding: i) frequent itemsets and ii) ARs from frequent itemsets. Finally, the ARs are selected according to measures such as support and confidence [23], [24]. Consider the AR $A \rightarrow B$, then we have the following measures.

The support of A (an itemset), $Support(A)$, is the number of transactions (sales) containing A divided by the total number of transactions [25].

***Support*:** the support of $A \rightarrow B$ is equal to the support of $A \cup B$ [23] and is expressed as $Support(A \cup B)$.
***Confidence*:** the confidence of $A \rightarrow B$ is expressed as follows [25]:

$$Confidence(A \rightarrow B) = Support(A \cup B)/Support(A) \qquad (1)$$

There are different algorithms to generate ARs, such as Apriori [24] and FP-growth [26], [27]. Any ARs algorithm must find the same set of ARs, although their computational efficiencies and memory requirements may be different.

The Apriori algorithm has two stages. In the first one, all the frequent itemsets are found. The steps of this stage are as follows [28]: All the transactions (e.g., sales) are analyzed to generate the candidate itemset C_1. That is, from the transactions, sets of one item (*1-itemsets*) named C_1 are generated. Then, the 1-itemsets that meet a minimum support threshold are selected, which are called frequent itemsets FI_1. From FI_1, subsets of two candidate items (*2-itemsets*) C_2 are generated and those that meet the minimum support threshold are selected and are called frequent itemsets FI_2. We continue similarly with $FI_2, \ldots, FI_k$ and this stage stops when no more subsets can be generated.

In the second stage, the ARs are generated from each of the frequent itemsets, for which the following algorithm is applied [28]:

1. For each frequent itemset FI_i ($i > 1$) obtain all non-empty subsets SFI of FI_i.
2. For each SFI of FI_i, create the AR: $SFI \rightarrow (FI_i - SFI)$.
3. Discard all ARs that do not meet a given minimum confidence threshold.

TABLE III
SOME STATISTICS FOR THE PRODUCTS OF TABLE II AFTER CLASSIFYING THEM INTO THE SUBCLASSES ACCORDING TO THE TREE IN FIG. 1

	Subclass								
	Processed liquid milk	Apples	Biscuits	Soap	Oranges	Brooms	Bottled waters	Meat of pigs, fresh or chilled	Cheese from milk of cattle, fresh or processed
Total number of units of the subclass in all sales	24	46	17	8	6	10	6	3	2
Total number of sales in which the subclass appears	24	6	17	8	3	6	4	3	2
% of sales in which subclass appears	100%	25%	70.833%	33.333%	12.5%	25%	16.667%	12.5%	8.333%

The names of the following subclasses (from UN DESA) have been abbreviated: **Biscuits =** Gingerbread and the like; sweet biscuits; waffles and wafers, **Soap =** Soap organic surface - active products and preparations for use as soap, **Brooms =** Brooms, brushes, hand-operated mechanical floor sweepers (not motorized), mops and feather dusters; prepared knots and tufts for broom or brush making; paint pads and rollers; squeegees (other than roller squeegees), and **Bottled waters =** Bottled waters, not sweetened or flavored.

IV. Proposed Algorithm

Next, we propose an algorithm that, from the sales history of products in a supermarket, generates one or more layouts. In each layout the same product (or product category) could appear in several departments of the supermarket, e.g., milk in the Bakery department and in the Fruits department. The inputs to the algorithm are described below. The definitions of the sets *PC*, *P*, and *S* are adapted from [29].

Let $PC = \{pc_1, pc_2, ..., pc_n\}$ be the set of nodes of a tree (a product category tree) where pc_i represents a product category. $pc_i = (pc_{id}, pc_{name}, pc_{parent})$, where $pc_{id} \in$ I+ is the unique identifier of each product category, pc_{name} (a string) is the name (also unique in *PC*) of the product category, and pc_{parent} is the identifier (pc_{id}) of a node $pc_j \in PC$ ($pc_i \neq pc_j$), i.e., pc_j is the parent product category of pc_i (if pc_i is the root node then pc_i has no parent). The parent product category of a node cannot be one of its descendant nodes, i.e., this is a hierarchical structure.

Let $P = \{p_1, p_2, ..., p_q\}$ be a set of products. $p_i = (p_{id}, p_{name}, p_{parent})$, where $p_{id} \in I^+$ is the unique identifier of each product, p_{name} (a string) is the name (also unique in P) of each product, p_{parent} is the identifier (pc_{id}) of a node $pc_i \in PC$, i.e., the product category (parent) to which p_i belongs.

Let $S = \{s_1, s_2, ..., s_n\}$ be a set of sales. $s = (s_{id}, D)$, where $s_{id} \in I^+$ is the unique identifier of each sale and $D = \{d_1, d_2, ..., d_m\}$ is a set of details. $d_j = (d_{product}, d_w)$, where $d_{product}$ is the identifier (p_{id}) of a product $p_i \in P$ and $d_w \in I^+$ is the number of units sold of p_i. There cannot be in D two details d_j, $d_i, i \neq j$, with the same $d_{product}$, i.e., there are not in D two details in the same sale that have the same product.

Example: $S = \{s_1, s_2, s_3\}$, where $s_1 = (1, \{(2, 20), (5, 2), (6, 7)\})$, $s_2 = (2, \{(8, 3), (5, 9), (7, 3)\})$, and $s_3 = (3, \{(2, 11)\})$.

Let $minssupp$ and $minconf \in R^+$ be the thresholds of support and confidence respectively that an AR must meet to be considered relevant to the analyst.

Let $CR = \{cr_1, cr_2, ..., cr_n\}$ be a set of constraints applied to the product category tree where cr_i is a constraint. $cr_i = (cr_{id}, cr_{product\ category\ 1}, cr_{product\ category\ 2})$, where $cr_{id} \in I^+$ is the unique identifier of each constraint, $cr_{product\ category\ 1}$ is the name (pc_{name}) of a node $pc_i \in PC$ of the product category that cannot be physically close (e.g., in the same department) to the product category $cr_{product\ category\ 2}$ which is the name (pc_{name}) of a node $pc_j \in PC$. We only consider constraints of this kind. Thus, e.g., we do not consider constraints involving three (or more) product categories pc_1, pc_2, and pc_3 such as: pc_1 can be close to pc_2 and pc_1 can be close to pc_3 but pc_1 cannot simultaneously be close to pc_2 and pc_3.

Considerations about the constraints:

- Equivalent constraints: Constraints $(cr_{id1}, cr_{product\ category\ 1}, cr_{product\ category\ 2})$ and $(cr_{id2}, cr_{product\ category\ 2}, cr_{product\ category\ 1})$ are equivalent. For example, the constraints (1, *Dairy products and egg products,Pome fruits and stone fruits*) and (2, *Pome fruits and stone fruits*, *Dairy products and egg products*) are equivalent.
- Constraint of different categories: Given a constraint $(cr_{id}, cr_{product\ category\ 1}, cr_{product\ category\ 2})$ then $cr_{product\ category\ 1} \neq cr_{product\ category\ 2}$. For example, the constraint: (1, *Dairy products and egg products, Dairy products and egg products*) is invalid.
- Constraint within the same branch of the product category tree: Given a constraint $(cr_{id}, cr_{product\ category\ 1}, cr_{product\ category\ 2})$ then $cr_{product\ category\ 1}$ cannot be a descendant of $cr_{product\ category\ 2}$ or $cr_{product\ category\ 2}$ cannot be a descendant of $cr_{product\ category\ 1}$. For example, the constraints (1, *Dairy products and egg products, Processed liquid milk*) and (1, *Processed liquid milk, Dairy products and egg products*) are invalid. Therefore, the product category corresponding to the root node, for which pc_{name} = *All products*, cannot be part of any constraint. For example, the constraint (1, *All products* , *Processed liquid milk*) is invalid.

Algorithm

Step 1: ARs

From S (the set of sales) the ARs are generated using the Apriori algorithm. Then we apply the methodology proposed in [17]: the set of ARs generated is subjected to weighting (*Wgain*) and utility (*Ugain*) constraints, and a combined utility weighted score (*UWScore*) is calculated for each AR [17]. Finally, a subset of ARs is selected based on the *UWScore.*

The steps are i) Mining of ARs using the Apriori algorithm, ii) Computation of *Wgain*, iii) Computation of *Ugain*, iv) Computation of *UWscore*, and v) Selection of relevant ARs based on *UWscore.*

Step 1.1: Mining of ARs using the Apriori algorithm

The Apriori algorithm generates k ARs: $AR = \{ar_1, ar_2, ..., ar_k\}$. Initially, the k ARs are ordered descendingly by confidence. The set of ordered ARs is $AR' = \{ar'_1, ar'_2, ..., ar'_k\}$.

Usually, the method used to determine the values for minimum support and confidence is based on the intuitiveness of the analyst [30]. If the support and confidence thresholds are too high, fewer rules are involved which leads to the loss of information [30].

By generating the ARs from the sales in Table II using the Apriori algorithm with a minimum support of 0.05, a minimum confidence of 0.5, and sorted descendingly by confidence, we obtain the ARs of Table IV (we show only the first six ARs). If there are ties in confidence, the ARs are sorted descendingly by support.

TABLE IV
ARs generated from Table II, sorted descendingly by confidence

#AR	LHS	RHS	Support	Confidence
1	{Biscuits}	{Processed liquid milk}	0.708	1
2	{Soap}	{Processed liquid milk}	0.333	1
3	{Biscuits, Soap}	{Processed liquid milk}	0.292	1
4	{Brooms}	{Processed liquid milk}	0.25	1
5	{Apples}	{Processed liquid milk}	0.25	1
6	{Bottled waters}	{Processed liquid milk}	0.167	1

Step 1.2: Computation of the measure Wgain

From the set AR', we select the first AR ar'_1, i.e., $\{Biscuits\} \rightarrow \{Processed\ liquid\ milk\}$. For each of the items (product categories) j of ar'_1, we calculate *Wgain* as follows:

$$Wgain_j = \sum_{i=1}^{|S|} d_{w(i,j)} \tag{2}$$

Where $d_{w(i,j)}$ represents the NUSP of product category j in sale i.

By calculating *Wgain* for the first five ARs of Table IV we obtain Table V. For example, *Wgain* = 17 for Biscuits. This means that the total sum of units of Biscuits is 17 considering all sales (S).

TABLE V
WGAIN OF THE FIRST FIVE ARS OF TABLE IV.

#AR	AR	Product category	Wgain
1	{Biscuits} → {Processed liquid milk}	Biscuits	17
		Processed liquid milk	24
2	{Soap} → {Processed liquid milk}	Soap	8
		Processed liquid milk	24
3	{Biscuits, Soap} → {Processed liquid milk}	Biscuits	17
		Soap	8
		Processed liquid milk	24
4	{Brooms} → {Processed liquid milk}	Brooms	10
		Processed liquid milk	24
5	{Apples} → {Processed liquid milk}	Apples	46
		Processed liquid milk	24

Step 1.3: Computation of the measure Ugain

Similarly, to calculate $Ugain$, we select the first AR ar'_1 from AR'. For each product category j of ar'_1 we calculate $Ugain$, which is based on the $Ufactor$ and the utility value of each product category. For simplicity, here we consider the utility of each product category as a constant c, i.e., they all have the same utility. For example, if we assume a utility $c = 1$ we obtain:

$$Ufactor = \frac{1}{\sum_{i=1}^{q} 1} = 1/q \tag{3}$$

Where q is the total number of distinct product categories.

$Ugain_j$ is calculated as follows:

$$Ugain_j = c * Ufactor \tag{4}$$

Since we are assuming an equal utility for all product categories, $c = 1$, then:

$$Ugain_j = Ufactor \tag{5}$$

For example, when calculating $Ugain$ for *Biscuits* with $q = 9$ we obtain:

$$Ugain_{Biscuits} = \frac{1}{\sum_{i=1}^{9} 1} = \frac{1}{9} \tag{6}$$

Step 1.4: Computation of UWscore from Wgain and Ugain

From the $Wgain$ and $Ugain$ measures calculated for each product category of an AR ar, we calculate a consolidated value called $UWScore$ for ar as follows:

$$UWScore_{ar} = \frac{\sum_{j=1}^{|ar|} (Wgain)_j * (Ugain)_j}{|ar|} \tag{7}$$

Where $|ar|$ is the number of product categories of ar.

As $Ugain_j = Ufactor$ and is the same for all product categories then:

$$UWScore_{ar} = Ufactor * \frac{\sum_{j=1}^{|ar|} (Wgain)_j}{|ar|} \tag{8}$$

The constant $Ufactor$ can be omitted since the effect is the same for all elements of the AR, then:

$$UWScore_{ar} = \frac{\sum_{j=1}^{|ar|} (Wgain)_j}{|ar|} \tag{9}$$

This formula represents the combined utility of ar based on the weighted NUSP and utility. For example, for AR ar_5: $\{Apples\} \rightarrow \{Processed\ liquid\ milk\}$ we obtain:

$$UWScore_{ar_5} = \frac{\sum_{j=1}^{|ar_5|} (Wgain)_j}{|ar_5|} = \frac{\sum_{j=1}^{2} (Wgain)_j}{2} = \frac{46 + 24}{2} = 35 \tag{10}$$

The processes to obtain *Wgain*, *Ugain*, and *UWScore* are repeated for the other ARs of AR'. The ARs are sorted descendingly by *UWScore*.

Step 1.5: Selection of ARs based on the UWscore

From the set of ARs sorted descendingly by $UWScore$, we select those that exceed a minimum threshold for $UWScore$.

Step 2: Selection of ARs with relevant support and confidence

In this step, with the ARs obtained from the previous step, the analyst can optionally apply a second filter with new $suppmin$ and $minconf$ thresholds, so that he/she can set limits regarding the total number of rules to be obtained [30].

Step 3: Selection of ARs that meet the constraints

Let us assume that there are constraints for the arrangement of the products in the supermarket departments, e.g., the products of the *Fruits and nuts product category* should not be close to the products of the *Soap* product category.

In this step, from the ARs of the previous step, we discard those whose product categories or their descendants violate any of the constraints of the set $CR = \{cr_1, cr_2, \ldots, cr_n\}$. For this, we calculate for each constraint cr_i all the descendants of its product categories, as follows:

$$DESCcr_i = \{[cr_{product\ category\ 1}, descendants(cr_{product\ category\ 1})], [cr_{product\ category\ 2}, descendants(cr_{product\ category\ 2})]\} \tag{11}$$

For example, if $CR = \{cr_1, cr_2, cr_3\}$, where:

- cr_1 = (1, "*Processed liquid milk cream and whey*", "*Soap and detergents, perfume and toilet preparations*").
- cr_2 = (2, "*Apples*", "*Soap and detergents, perfume and toilet preparations*").
- cr_3 = (3, "*Biscuits*", "*Soap and detergents, perfume and toilet preparations*").

When calculating the descendants for cr_1 we obtain:

$$DESCcr_1 = \left\{ \begin{array}{c} [\text{"Processed liquid milk cream and whey"}, \text{"Processed liquid milk"}, \\ \text{"Processed liquid milk"}, \text{"Lactose − free milk"}, \text{"Whole milk"}], \\ [\text{"Soap, cleaning preparations, perfumes and toilet preparations"}, \\ \text{"Soap and detergents, perfume and toilet preparations"}, \\ \text{"Soap; organic surface − active products and preparations for use as soap;} \\ \text{paper, wadding, felt and nonwovens, impregnated, coated or covered with soap or detergent"}] \end{array} \right\} \quad (12)$$

Note that in the descendants of *Processed liquid milk cream and whey*, *Processed liquid milk* is repeated, this is because according to the UN DESA classification, the class and subclass have the same name. Similarly, cr_2 and cr_3 are calculated. For example, the AR $\{Soap\} \rightarrow \{Processed\ liquid\ milk\}$ violates cr_1, therefore; it is discarded, since the product category Soap and its descendants cannot be close to the product category *Processed liquid milk cream and whey* and its descendants. Remember *Soap* is the abbreviated name for "*Soap organic surface - active products and preparations for use as soap*", see Table III.

Step 4: Generate clusters of nearby products

From the set of ARs found in the previous step, we group the ARs to obtain the clusters, i.e., the supermarket departments with their respective product categories.

Clustering is the process of examining a collection of points and grouping them according to some measure of distance [31]. The objective is that the points in the same cluster have a distance less than a given threshold [32]. To obtain the clusters we apply the hierarchical clustering algorithm. The algorithm starts with each point forming its own cluster and is summarized as follows [32]:

WHILE it is not time to stop DO
    Pick the "best" two clusters to merge
    Combine those two clusters into one cluster
END;

We chose as a stopping rule that the algorithm ends when there are g clusters (departments). Consider the set of ARs $\{ar_5, ar_{31}, ar_7, ar_{32}, ar_1, ar_{30}, ar_4, ar_6, ar_{10}\}$ of Table VI.

TABLE VI
ARs RESULTING EXAMPLE FOR CLUSTERING

#AR	AR
5	{Apples} → {Processed liquid milk}
31	{Apples} → {Biscuits}
7	{Apples,Biscuits} → {Processed liquid milk}
32	{Apples,Processed liquid milk} → {Biscuits}
1	{Biscuits} → {Processed liquid milk}
30	{Processed liquid milk} → {Biscuits}
4	{Brooms - brushes} → {Processed liquid milk}
6	{Bottled waters} → {Processed liquid milk}
10	{Oranges} → {Processed liquid milk}

As an example, with $g = 4$, we start by assigning each AR to a cluster. The stopping rule indicates that four clusters must be reached, currently there are nine (since there are nine ARs), then the algorithm continues and the two "best" clusters must be chosen to be combined, for this the distances between each pair of them are calculated and the pair with the smallest distance is combined.

To calculate the distances between ARs we use a symmetric distance matrix D [33], which contains the distances for all pairs of ARs. In this matrix an element $D(i,j)$ corresponds to the distance between the ARs ar_i and ar_j denoted $dist(ar_i, ar_j)$. Note that $dist(ar_i, ar_j) = dist(ar_j, ar_i)$ and $dist(ar_i, ar_i) = 0$.

The distance between two ARs is calculated as follows [33]:

Let ARs ar_1: $A_1 \rightarrow B_1$ and ar_2: $A_2 \rightarrow B_2$, then:

$$dist(ar_1, ar_2) = \omega_0 * Jaccard(A_1 \cup B_1, A_2 \cup B_2) + \omega_1 * \alpha + \omega_2 * \beta \quad (13)$$

Where:

$$Jaccard(A, B) = \frac{|A \cup B| - |A \cap B|}{|A \cup B|} \quad (14)$$

$$\alpha = 1 - \frac{Support_{ar_1} + Support_{ar_2}}{2} \quad (15)$$

$$\beta = 1 - \frac{Confidence_{ar_1} + Confidence_{ar_2}}{2} \quad (16)$$

The Jaccard distance measures the difference between two sets by considering the proportion of different elements between the two sets with respect to all the elements of the two sets [28]. To include support and confidence in the calculation of the distances, parameters α and β [33] are defined, in which supports and confidences are averaged accordingly. ω_0, ω_1, and ω_2 are weights, which allow the analyst to give more importance to certain factors in the formula.

As an example, $dist(ar_5, ar_{31})$ is calculated with weights $\omega_0 = 0.6$, $\omega_1 = 0.2$, and $\omega_2 = 0.2$. We have $ar_5 = \{Apples\} \rightarrow \{Processed\ liquid\ milk\}$ and $ar_{31} = \{Apples\} \rightarrow \{Biscuits\}$.

$$dist(ar_5, ar_{31}) = 0.6 * Jaccard(\{Apples\} \cup \{Processed\ liquid\ milk\}, \{Apples\} \cup \{Biscuits\}) + 0.2 * \alpha + \omega_2 * \beta \quad (17)$$

Where:

$$Jaccard(\{Apples\} \cup \{Processed\ liquid\ milk\}, \{Apples\} \cup \{Biscuits\}) = \frac{3 - 1}{3} = \frac{2}{3} \quad (18)$$

Next, we calculate α and β:

$$\alpha = 1 - \frac{Support_{ar_5} + Support_{ar_{31}}}{2} = 1 - \frac{0.25 + 0.167}{2} = 0.792 \quad (19)$$

$$\beta = 1 - \frac{Confidence_{ar_5} + Confidence_{ar_{31}}}{2} \quad (20)$$

$$= 1 - \frac{1 + 0.667}{2} = 0.165$$

Then

$$dist(ar_5, ar_{31}) = 0.6 * \left(\frac{2}{3}\right) + 0.2 * 0.792 + 0.2 * 0.165 = 0.591 \quad (21)$$

Similarly, using the same values of ω_0, ω_1, and ω_2, we calculate the distances for all pairs of ARs and obtain the matrix D. The two ARs with the smallest distance in D are selected and combined into one cluster, thus ending the first iteration of the algorithm.

As the current number of clusters is eight and the stopping rule indicates that the process must stop when there are four clusters, the best clusters to combine must be selected again, for this, the new matrix D must be calculated considering the new cluster.

In the calculation of the new distances with respect to a cluster (formed by one or more ARs) we proceed as follows. To calculate the distance from a cluster clu_1 that has one or more ARs to another cluster clu_2 (which also has one or more ARs), we calculate all the distances between the ARs of clu_1 with respect to the ARs of clu_2 and we choose the maximum distance (this is called complete linkage distance [34]), then we combine the two clusters that have the smallest complete linkage distance, that is:

$$clusterDist_{clu_1,\ clu_2} = max\,\{\, dist(ar_i, ar_j),\ where\ ar_i\ is\ in\ cluster\ clu_1\ and\ ar_j\ is\ in\ cluster\ clu_2\,\} \quad (22)$$

Step 5: Layout design

Finally, we generate the layout for the supermarket. Assuming clusters clu_1 = {ar_{10}}, clu_2 = {ar_4, ar_6}, clu_3 = {ar_{31}}, and clu_4 = {ar_5, ar_1, ar_{30}, ar_7, ar_{32}}, then by representing the LHS and RHS of the ARs in a single set we obtain clu_1 = {Oranges, Processed liquid milk}, clu_2 = {Brooms, Processed liquid milk, Bottled waters}, clu_3 = {Apples, Biscuits}, and clu_4 = {Apples, Processed liquid milk, Biscuits}. In Fig. 2 we show the supermarket departments with the clusters of product categories.

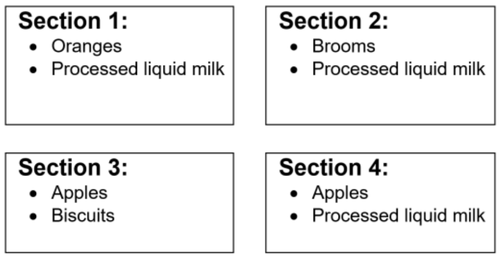

Fig. 2. Supermarket layout proposal

In the above layout we can see that there are product categories that are found in several departments, e.g., Processed liquid milk is found in departments 1, 2, and 4

Figure 3 summarizes the steps of our algorithm.

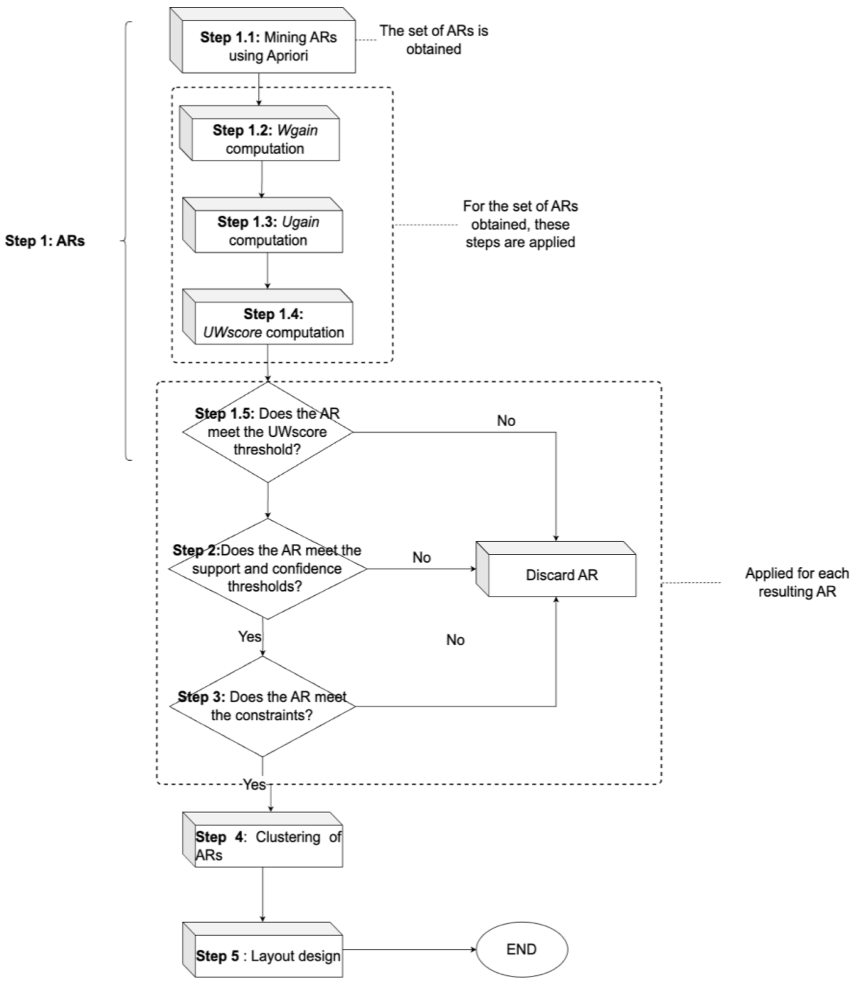

Fig. 3. Summary of the algorithm steps

If initially there is no sales data (i.e., "a start cold"), e.g., a new supermarket, the traditional approach described in [2], [6] can be used. Then, when there are sales, the algorithm can be applied and one of the generated layouts can be implemented. Periodically, the algorithm can be applied, and changes in the layout (evolution of the layout) can be obtained according to the new sales.

Additionally, the analyst may determine that some products (or product categories) will not be included in the algorithm because they have a fixed location, e.g., books, magazines, and candies are usually placed near the cash registers.

V. Experiments

In this section we apply our algorithm. We took a sample of 9835 sales from a supermarket [35], from which 200 were selected to illustrate our algorithm. This number of sales was selected due to the complexity of manually classifying the products of each transaction following the UN DESA classification. In Table VII we show five sales. In *https://docs.google.com/spreadsheets/d/1qpEY5b3EgqvDG2zXV2aE2jY7CTuq-JEZ/edit#gid=336827795*, we show the complete list.

TABLE VII
Sample of 5 Sales

#	Sale
1	Citrus fruit, Semi-finished bread, Margarine, Ready soups
2	Tropical fruit, Yogurt, Coffee
3	Whole milk
4	Pip fruit, Yogurt, Cream cheese, Meat spreads
5	Other vegetables, Whole milk, Condensed milk, Long life bakery product

Note that the data sample does not include the NUSP. The products of the sales were classified according to the UN DESA. The 200 sales have 118 different products, each one was manually classified into *section*, *division*, *group*, *class*, and *subclass*. In Table VII we show the classification result for five products and in *https://docs.google.com/spreadsheets/d/1sVZe_v8gAWAHvLEdIWQ-NhB4U-JJ587f/edit#gid=858255698*, we show the classification of the 118 products.

For the experiments we used the group level classification, i.e., *a group corresponds to a product category* according to our conventions, as input to the algorithm. Next, we show the results of each step.

A. Step 1: ARs

Step 1.1: AR mining using the Apriori algorithm

From the sales of *https://docs.google.com/spreadsheets/d/1qpEY5b3EgqvDG2zXV2aE2jY7CTuq-JEZ/edit#gid=336827795*, we generated the ARs using the Apriori algorithm, with a $suppmin$ = 0.08 and $minconf$ = 0.3 and obtained 32 ARs. By sorting by confidence, we show the ARs in Table IX. In *https://docs.google.com/spreadsheets/d/1BUMWtBuu3QuOIIovZl7alHRuTm59Nobs/edit#gid=1004684258*, we show all the ARs.

TABLE IX
Sample of five ARs sorted descendingly by confidence, generated from the 200 sales

#AR	LHS	RHS	Support	Confidence
1	{Fruits and nuts}	{Other dairy products}	0.130	0.542
2	{Meat and meat products}	{Bakery products}	0.140	0.528
3	{Processed liquid milk, cream and whey}	{Other dairy products}	0.160	0.508
4	{Vegetables}	{Bakery products}	0.120	0.480
5	{Other dairy products}	{Processed liquid milk, cream and whey}	0.160	0.478

Step 1.2: Computation of the measure Wgain

To calculate the *Wgain*, the NUSP in the 200 sales is required. Since the sales in the sample do not include the NUSP, pseudo-random integers in the range of 1 - 50 were generated to obtain it. In Table X we show the values obtained for each product category.

TABLE X
Wgain obtained for each product category

Product category	Wgain
Fruits and nuts	26
Meat and meat products	30
Processed liquid milk, cream and whey	38
Vegetables	26
Other dairy products	49
Soft drinks; bottled mineral waters	25
Bakery products	13

Next, we calculated the *Wgain* of each product category

TABLE VIII
Sample classification of products according to the UN DESA

Name	Section	Division	Group	Class	Subclass
Citrus fruit	Agriculture, forestry and fishery products	Products of agriculture, horticulture and market gardening	Fruits and nuts	Citrus fruits	Oranges
Tropical fruit	Agriculture, forestry and fishery products	Products of agriculture, horticulture and market gardening	Fruits and nuts	Tropical and subtropical fruits	Bananas
Whole milk	Food products, beverages and tobacco; textiles, apparel and leather products	Dairy products and egg products	Processed liquid milk, cream and whey	Processed liquid milk	Processed liquid milk
Pip fruit	Agriculture, forestry and fishery products	Products of agriculture, horticulture and market gardening	Fruits and nuts	Pome fruits and stone fruits	Apples

of the ARs. In Table XI we present the results for the first five ARs. In *https://docs.google.com/spreadsheets/d/1BUMWtBuu3QuOIIovZl7alHRuTm59Nobs/edit#gid=1004684258*, we present the results for the rest (we used the label "*ANY*" for the products that could not be classified in a product category (i.e., group) of those proposed in UN DESA).

TABLE XI
WGAIN OF THE FIRST FIVE ARS OF TABLE IX

#AR	AR	Product category	Wgain
1	{Fruits and nuts} → {Other dairy products}	Fruits and nuts	26
		Other dairy products	49
2	{Meat and meat products} → {Bakery products}	Meat and meat products	30
		Bakery products	13
3	{Processed liquid milk, cream and whey} → {Other dairy products}	Processed liquid milk, cream and whey	38
		Other dairy products	49
4	{Vegetables} → {Bakery products}	Vegetables	26
		Bakery products	13
5	{Other dairy products} → {Processed liquid milk, cream and whey}	Other dairy products	49
		Processed liquid milk, cream and whey	38

Step 1.3: Computation of the measure Ugain

As explained in step 1.3 as we do not consider the utility of the products, then $Ugain_j$ is the same for all product categories. Therefore, we omit its calculation as it does not affect the *UWScore*, see next step.

Step 1.4: Computation of UWscore from Wgain and Ugain

From the previous steps we calculate the *UWScore*. In Table XII we present the first five ARs. In *https://docs.google.com/spreadsheets/d/1BUMWtBuu3QuOIIovZl7alHRuTm59Nobs/edit#gid=1004684258*, we present the result for all the ARs sorted descendingly by *UWscore*.

Step 1.5: Selection of ARs based on the UWscore

We considered a threshold of $UWScore \geq 20$ and we obtained a total of 27 rules.

Step 2: Selection of ARs with relevant support and confidence

We chose the ARs from the previous step that meet the following thresholds: $suppmin = 0.1$ and $minconf = 0.4$. Note that we used support and confidence thresholds different from those in subsection 4.1, where $suppmin$ = 0.08 and $minconf$ = 0.3. The value of both thresholds was increased to further limit the number of ARs to be obtained.

We show the resulting ARs in Table XIII.

Step 3: Selection of ARs that meet the constraints

In the results of the previous step, see Table XIII, we considered that there are no product categories that by traditional or sanitary conceptions should not be in the same department. Indeed, all product categories are food.

TABLE XIII
THE FIVE ARS WITH THE HIGHEST *UWSCORE*.

#AR	AR	Attribute	Wgain	|AR|	UW-score
3	{Processed liquid milk, cream and whey} → {Other dairy products}	Processed liquid milk, cream and whey	38	2	43.5
		Other dairy products	49		
5	{Other dairy products} → {Processed liquid milk, cream and whey}	Other dairy products	49	2	43.5
		Processed liquid milk, cream and whey	38		
24	{Meat and meat products} → {Other dairy products}	Meat and meat products	30	2	39.5
		Other dairy products	49		
1	{Fruits and nuts} → {Other dairy products}	Fruits and nuts	26	2	37.5
		Other dairy products	49		
14	{Other dairy products} → {Fruits and nuts}	Other dairy products	49	2	37.5
		Fruits and nuts	26		

TABLE XII
ARS THAT MEET THE THRESHOLDS $minsupp = 0.1$ AND $minconf = 0.4$

#AR	AR	Attribute	Wgain	|AR|	UW-score	Support	Confidence
3	{Processed liquid milk, cream and whey} → {Other dairy products}	Processed liquid milk, cream and whey	38	2	43.5	0.160	0.508
		Other dairy products	49				
5	{Other dairy products} → {Processed liquid milk, cream and whey}	Other dairy products	49	2	43.5	0.160	0.478
		Processed liquid milk, cream and whey	38				
1	{Fruits and nuts} → {Other dairy products}	Fruits and nuts	26	2	37.5	0.130	0.542
		Other dairy products	49				
9	{Meat and meat products} → {Processed liquid milk, cream and whey}	Meat and meat products	30	2	34	0.110	0.415
		Processed liquid milk, cream and whey	38				
11	{Vegetables} → {Processed liquid milk, cream and whey}	Vegetables	26	2	32	0.100	0.400
		Processed liquid milk, cream and whey	38				
8	{Fruits and nuts} → {Meat and meat products}	Fruits and nuts	26	2	28	0.105	0.438
		Meat and meat products	30				
6	{Processed liquid milk, cream and whey} → {Bakery products}	Processed liquid milk, cream and whey	38	2	25.5	0.150	0.476
		Bakery products	13				

Step 4: Clustering of ARs

In this step we generate two layouts. We arbitrarily chose g = 7 and g = 4; this choice allows us to observe how the product distribution in the layouts is affected by decreasing or increasing the number of departments.

In the calculation of the distances between the ARs, we used the weights $\omega_0 = 0.6$, $\omega_1 = 0.2$, and $\omega_2 = 0.2$. These weights were selected to give more importance to the Jaccard distance between rule items, as opposed to support and confidence, but any value can be chosen at the analyst's criteria.

For g = 7 we obtained the clusters $clu_1 = \{ar_8\}$, $clu_2 = \{ar_{11}\}$, $clu_3 = \{ar_5\}$, $clu_4 = \{ar_9\}$, $clu_5 = \{ar_3, ar_1\}$, $clu_6 = \{ar_2\}$, and $clu_7 = \{ar_6, ar_{10}\}$.

For g = 4 we obtained the clusters $clu_1 = \{ar_8\}$, $clu_2 = \{ar_{11}, ar_5, ar_9\}$, $clu_3 = \{ar_3, ar_1\}$, and $clu_4 = \{ar_2, ar_6, ar_{10}\}$

Step 5: Layout design

In this step, from the clusters obtained in the previous step we generated the layout proposals.

By representing the LSH and RHS of the ARs obtained in a set we obtained.

For g = 7:

- clu_1 = {*Fruits and nuts*, *Meat and meat products*}
- clu_2 = {*Vegetables*, *Processed liquid milk, cream and whey*}
- clu_3 = {*Other dairy products*, *Processed liquid milk, cream and whey*}
- clu_4 = {*Meat and meat products*, *Processed liquid milk, cream and whey*}
- clu_5 = {*Processed liquid milk, cream and whey*, *Other dairy products*, *Fruits and nuts*}
- clu_6 = {*Meat and meat products*, *Bakery products*}
- clu_7 = {*Processed liquid milk, cream and whey*, *Bakery products*}

We show the proposed layout for g = 7 in Fig. 4.

For g = 4:

clu_1 = {*Fruits and nuts*, *Meat and meat products*}

clu_2 = {*Vegetables*, *Processed liquid milk, cream and whey*, *Other dairy products*, *Meat and meat products*}

clu_3 = {*Processed liquid milk, cream and whey*, *Other dairy products*, *Fruits and nuts*}

clu_4 = {*Meat and meat products*, *Bakery products*, *Processed liquid milk, cream and whey*}

We show the proposed layout for g = 4 in Fig. 5.

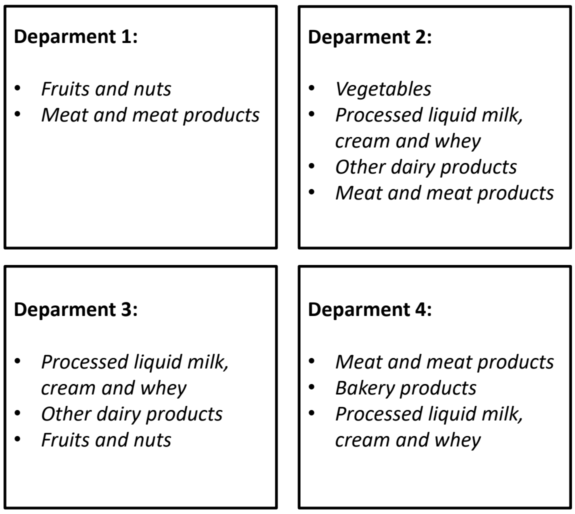

Fig. 5. Proposed layout for the supermarket for g = 4

As the data sample was only 200 sales, experiments with more sales are required to obtain more reliable results. For example, we expected that by traditional conceptions *Fruits and nuts* and *Vegetables* would be in the same department. However, in neither of the two layouts did this occur. Also, note that:

- Departments 1 in both layouts are equal.
- Department 5 of the layout for g = 7 is equal to department 3 of the layout for g = 4.
- Departments 2, 3, and 4 for g = 7 are equal to department 2 of the layout with g = 4.
- Departments 6 and 7 with g = 7 are equal to department 4 of the layout with g = 4.
- In both layouts there are product categories that appear in more than one department.

With a larger data sample, we could obtain results that possibly reflect some traditional conceptions or suggest non-

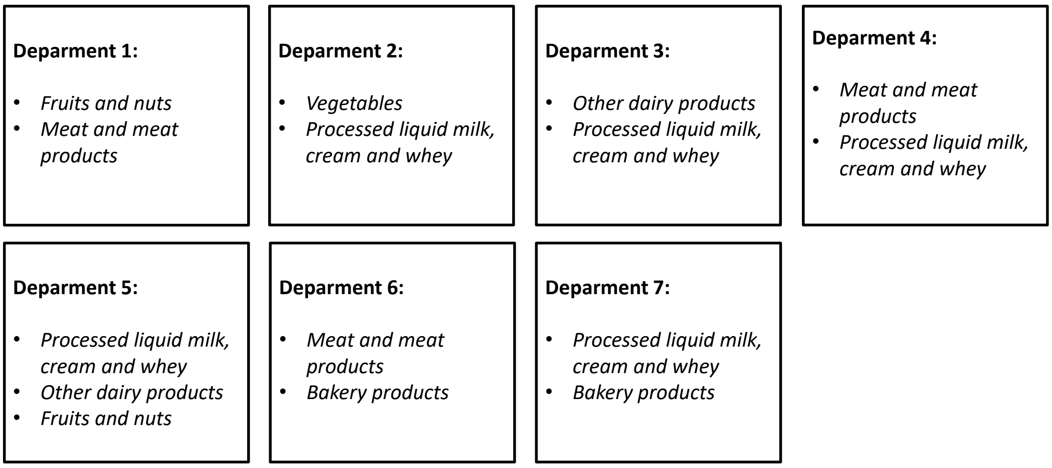

Fig. 4. Layout proposal for the supermarket for g = 7

intuitive layouts.

When comparing the layouts obtained with a traditional layout, such as the one shown in Fig. 6(a) of a Japanese supermarket (redrawing based on [37]), we can see that it has a single department for fruits and vegetables (Vegetables & Fruits); but in each of the layouts obtained by our proposal there are two departments that include fruits. Similarly, Fig. 6(b) shows a supermarket in the Philippines (redrawing based on [38]), where something similar is observed: a single department for the product categories. Finally, when comparing with the proposal of Cil [2], see Fig. 6(c) (redrawing based on [2]), we can see a similar situation, although, in his proposal some products of the department “Promotion products” may be shared with other departments.

Similarly, *Processed liquid milk, cream and whey* products are likely to be found in the Japanese supermarket only in the "General Foods" department as compared to our proposal's layouts where they are found in several departments. This is also true for the product category *Meat and meat products*, which is in three departments in each of our layouts.

VI. Conclusions

In this paper we proposed an algorithm to generate several layouts of a supermarket, i.e., which products or product categories should be located in each department, based on ARs in which we consider i) the NUSP for the generation of the ARs, ii) the use of a hierarchical structure for the classification of the products designed by the UN DESA, iii) a set of constraints between the products (or product categories) that indicate that they should not be located close to each other (in the same department), and iv) the generation of clusters of ARs. These clusters indicate which products (or product categories) should be in the same department. In the results we could notice that there were clusters (departments) that have product categories in common.

However, the data sample used in our experiments can be considered small (2% of the total data sample) and biased

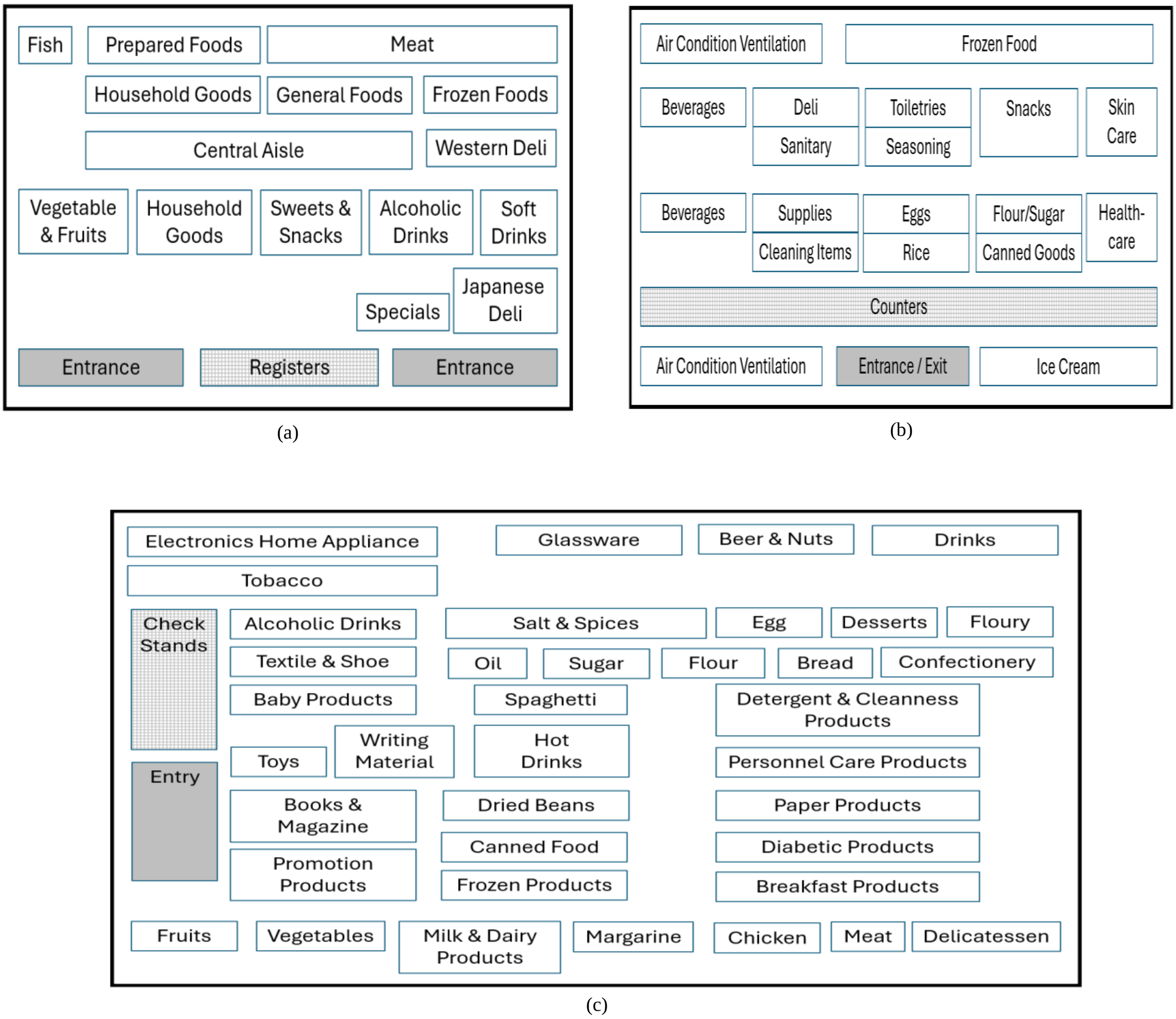

Fig. 6. Examples of supermarkets layouts: a) a Japanese supermarket (redrawing based on [37]), b) a grocery store layout in the Philippines (redrawing based on [38]), and c) a proposal for a supermarket layout (redrawing based on [2]).

results may have been found. For example, we expected to find that by traditional conceptions *Fruits and nuts* and *Vegetables* were in the same department. However, in the layout obtained this did not occur.

As future work, we hope to specialize our algorithm for the layout of other physical spaces, e.g., clothing stores, electronics stores, warehouses, among others. The algorithm can also be extended by considering the utility variable of the products, a factor that was omitted when calculating the *Ugain* measure. We also intend to develop a methodology in which measures are defined to evaluate which layout yields the highest profits. The discovery of rare patterns of items [36] is also an interesting line of work.

In addition, we hope to develop a visualization module so that the manager or an analyst in charge can: i) enter the supermarket map with the available departments, ii) load the sales to generate the layouts from the algorithm presented, and iii) generate the supermarket map indicating in each department which products (or product categories) should be located.

We also plan to develop a tool for classifying the products automatically in the UN DESA hierarchy. This operation is time consuming to perform manually and is challenging because there are product names that are not clear where to classify them, e.g., "Instant food products".

References

[1] D. M. Lewison, *Retailing*. New York: Macmillan College, 1994.

[2] 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, Aug. 2012.

[3] K. Buza, A. Buza, and P. B. Kis, "Towards better modeling of supermarkets," in *International Joint Conference on Computational Cybernetics and Technical Informatics*, Timisoara, 2010, pp. 499–503.

[4] A. Borges, "Toward a new supermarket layout: from industrial categories to one stop shopping organization through a data mining approach," Reims Management School, Reims, France, Tech. Rep., 2004.

[5] A. F. Arboleda and F. J. Moreno, "Análisis y oportunidades para el diseño de supermercados basado en reglas de asociación," in *Computación para el Desarrollo – XIV Congreso*, L. Bengochea Martínez, D. Meziat Luna, and A. López Poveda, Eds., Managua, 2021, pp. 57–67.

[6] G. R. Peterson, "Economics store layout and design," *Journal of Food Distribution Research*, vol. 2, no. 1, pp. 1–10, Jan. 1970.

[7] 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, Dec. 2005.

[8] S. Altuntas, "A novel approach based on utility mining for store layout: a case study in a supermarket," *Industrial Management & Data Systems*, vol. 117, no. 2, pp. 304–319, Mar. 2017.

[9] S. Altuntas and H. Selim, "Facility layout using weighted association rule-based data mining algorithms: evaluation with simulation," *Expert Systems with Applications*, vol. 39, no. 1, pp. 3–13, Jan. 2012.

[10] R. Agrawal, T. Imieliński, and A. Swami, "Mining association rules between sets of items in large databases," in *Proc. of the 1993 ACM SIGMOD Int. Conf. on Management of Data*, Washington D.C., 1993, pp. 207–216.

[11] Y. F. Wang, Y. L. Chuang, M. H. Hsu, and K. Huan-Chao, "A personalized recommender system for the cosmetic business," *Expert Systems with Applications*, vol. 26, no. 3, pp. 427–434, Apr. 2004.

[12] Department of Economic and Social Affairs. (2015, Jan 15) Central product classification (CPC) version 2.1 (Online). Available: https://unstats.un.org/unsd/classifications/Econ/Download/In Text/CPCv2.1_complete(PDF)_English.pdf

[13] A. Adhikari and P. R. Rao, "Association rules induced by item and quantity purchased," in *Database Systems for Advanced Applications: 13th Int. Conf.*, New Delhi, 2008, pp. 478–485.

[14] F. Alfiah, B. W. Pandhito, A. T. Sunarni, D. Muharam, and P. R. Matusin, "Data mining systems to determine sales trends and quantity forecast using association rule and CRISP-DM method," *International Journal of Engineering and Techniques*, vol. 4, no. 1, pp. 186–192, Feb. 2018.

[15] K. Preetham and V. S. Ananthanarayana, "Discovery of frequent itemsets based on minimum quantity and support," *International Journal of Computer Science and Security*, vol. 3, no. 3, pp. 216–225, Jun. 2009.

[16] S. Ibrahim and J. Revathy, "A novel quantity based weighted association rule mining," *International Journal of Engineering Inventions*, vol. 4, no. 3, pp. 33–38, Aug. 2014.

[17] P. S. Sandhu, D. S. Dhaliwal, and S. N. Panda, "Mining utility-oriented association rules: an efficient approach based on profit and quantity," *International Journal of the Physical Sciences*, vol. 6, no. 2, pp. 301–307, Jan. 2011.

[18] S. Halim, T. Octavia, and C. Alianto, "Designing facility layout of an amusement arcade using market basket analysis," *Procedia Computer Science*, vol. 161, no. 1, pp. 623–629, Apr. 2019.

[19] Y. L. Chen, K. Tang, R. J. Shen, and Y. H. Hu, "Market basket analysis in a multiple store environment," *Decision Support Systems*, vol. 40, no. 2, pp. 339–354, Aug. 2005.

[20] L. C. Annie and A. Kumar, "Market basket analysis for a supermarket based on frequent itemset mining," *International Journal of Computer Science Issues*, vol. 9, no. 5, pp. 257–264, Sep. 2012.

[21] M. A. Alan and A. R. Ince, "Use of association rule mining within the framework of a customer-oriented approach," *European Scientific Journal*, vol. 12, no. 9, pp. 81–99, Mar. 2016.

[22] R. Z. Farahani, M. Steadieseifi, and N. Asgari, "Multiple criteria facility location problems: a survey," *Applied Mathematical Modelling*, vol. 34, no. 7, pp. 1689–1709, Jul. 2010.

[23] R. Paper, M. Shweta and K. Garg, "Mining efficient association rules through apriori algorithm using attributes and comparative analysis of various association rule algorithms," *International Journal of Advanced Research in Computer Science and Software Engineering*, vol. 3, no. 6, pp. 306–312, Jun. 2013.

[24] R. Agrawal, R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th Int. Conf. VLDB*, Santiago de Chile, 1994, pp. 487–499.

[25] M. J. Zaki and M. Ogihara, "Theoretical foundations of association rules," in 3rd ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, Seattle, 2004, pp. 71-78.

[26] J. Han, J. Pei, and Y. Yin, "Mining frequent patterns without candidate generation," *ACM Sigmod Record*, vol. 29, no. 2, pp. 1–12, Jun. 2000.

[27] J. Han, J. Pei, Y. Yin, and R. Mao, "Mining frequent patterns without candidate generation: a frequent-pattern tree approach," *Data Mining and Knowledge Discovery*, vol. 8, no. 1, pp. 53–87, Jan. 2004.

[28] L. Futai, L. Hongquan, Z. Weifeng, and Z. Chenhao, "An improved distance metric clustering algorithm for association rules," *Journal of Physics: Conference Series*, vol. 1284, no. 1, pp. 1–6, Aug. 2019.

[29] Francisco Javier Moreno Arboleda, Giovanni Perez Ortega, and Jaime Alberto Guzman Luna, "Temporal Visual Profiling of Market Basket Analysis," IAENG International Journal of Computer Science, vol. 49, no.2, pp582-594, 2022

[30] E. Hikmawati, N. U. Maulidevi, and K. Surendro, "Minimum threshold determination method based on dataset characteristics in association rule mining," *Journal of Big Data*, vol. 8, no. 146, pp. 1–17, Dec. 2021.

[31] Omer Arifogullari, and Gunce Keziban Orman, "On Experimental Study of Hotel Clustering," Lecture Notes in Engineering and Computer Science: Proceedings of The International MultiConference of Engineers and Computer Scientists 2023, 5-7 July, 2023, Hong Kong, pp82-87

[32] J. Leskovec, A. Rajaraman, and J. D. Ullman, *Mining of Massive Datasets*. Cambridge: Cambridge University Press, 2014, ch. 7.

[33] S. Theodoridis and K. Koutroumbas, *Pattern Recognition.* Boston: Academic Press, 2009, ch. 13.

[34] Penn State Eberly Collage of Science. (2023, Feb 7). Agglomerative hierarchical clustering (Online). Available: https://online.stat.psu.edu/stat505/lesson/14/14.4

[35] Kaggle. (2019, Apr 5). Groceries market basket dataset (Online).

Available:
https://www.kaggle.com/datasets/irfanasrullah/groceries

[36] Jyothi Upadhya K, Thanishka Bolisetty, B Dinesh Rao, and Geetha M, "Negative Itemset Tree for Discovering Rare Patterns with Periodicity from Static Databases," Engineering Letters, vol. 31, no.3, pp882-895, 2023

[37] Y. Zuo, A. B. M. S. Ali, and K. Yada, “Consumer purchasing behavior extraction using statistical learning theory,” *Procedia Computer Science*, vol. 35, no. 4, pp. 1464–1473, Dec. 2014.

[38] M. F. Andrada, H. L. Prudenciano, and J. B. Reyes, “Layout design model for independent grocery stores in the Philippines,” in *Proc. 10th Int. Conf. on Industrial Engineering and Operations Management*, Dubai, 2020, pp. 1719–1731.