Hierarchical Reasoning and Knapsack Problem Modelling to Design the Ideal Assortment in Retail

Jocelyn Poncelet[1,2](✉), Pierre-Antoine Jean[1](✉), Michel Vasquez[1](✉), and Jacky Montmain[1](✉)

[1] EuroMov Digital Health in Motion, Univ Montpellier,
IMT Mines Ales, Alès, France
{jocelyn.poncelet,pierre-antoine.jean,michel.vasquez,
jacky.montmain}@mines-ales.fr

[2] TRF Retail, 116 allée Norbert Wiener, 30000 Nîmes, France

Abstract. The survival of a supermarket chain is heavily dependent on its capacity to maintain the loyalty of its customers. Proposing adequate products to customers is the issue of the store's assortment. With tens thousands of products on shelves, designing the ideal assortment is theoretically a thorny combinatorial optimization problem. The approach we propose includes prior knowledge on the hierarchical organization of products by family to formalize the ideal assortment problem into a knapsack problem. The main difficulty of the optimization problem remains the estimation of the expected benefits associated to changes in the product range of products' families. This estimate is based on the accounting results of similar stores. The definition of the similarity between two stores is then crucial. It is based on the prior knowledge on the hierarchical organization of products that allows approximate reasoning to compare any two stores and constitutes the major contribution of this paper.

Keywords: Optimal assortment in mass distribution · Semantic similarity measures · Knapsack problem

1 Introduction

Competition in large retailers is becoming increasingly intense; therefore, in order to satisfy fluctuating demand and customers' increasing expectations, deal with the competition and remain or become market leaders, retailers must focus on searching for sustainable advantages. The survival of a supermarket chain is heavily dependent on its capacity to maintain the loyalty of its customers [11,12]. Proposing adequate products to its customers is the issue of the store's assortment, *i.e.*, products offered for sale on shelves [10,13]. Moreover, retailers are faced to manage high stores' networks. This way, they use a common assortment shared in the stores' network to allow an easier management [14].

M.-J. Lesot et al. (Eds.): IPMU 2020, CCIS 1237, pp. 201–214, 2020.
https://doi.org/10.1007/978-3-030-50146-4_16

Therefore, stores share a common and centralised assortment [18] with some tolerated exceptions to take into account specific characteristics of stores in the network [16]. To improve their global performance, retailers aim to increase their knowledge on stores' specific characteristics in the network to suggest the optimal assortment for each store, *e.g.*, they try to identify the products that perform remarkably in some stores of the network to recommend them to other *similar* stores. However, the definition of *similar* stores is not so obvious: it can be related to the localisation of stores, their assortments on shelves, their revenues, their format, *e.g.*, Hypermarket, Supermarket... [11]. This concept of similarity plays a central role in this contribution.

In this article we address the question of the ideal assortment in supermarkets. To better understand the complexity of the task, it must be remembered that some hypermarkets offer up to 100,000 products [16]. Defining the ideal assortment in a department store consists in selecting this set of products. More formally, this thorny problem corresponds to an insoluble combinatorial optimization problem. In practice, decisions are made locally by a category manager, while the problem of the ideal assortment should correspond to a global decision at the store level. To tackle this question, retailers have prior knowledge available [17]. Indeed, department store are organized into categories, *e.g.*, `food`, `household products`, `textiles`, etc. These categories are themselves divided into families or units of need (*e.g.* `textiles` category is derived into `woman`, `man` and `child` sections and so on). This hierarchical organization of products makes it possible to reason about families of products, structure the decision and thus avoid combinatorial explosion. Most of the time, a hypermarket cannot choose a single product to increase its offer. Indeed, this additional item necessarily belongs to a level of assortment or product line, generally in adequacy with the size or the location of the store: choosing a product requires to take all products associated to the same level of assortment [16,18]. For example, if a store offers a soda section, it can be satisfied with a minimum offer, *e.g.*, `Coca-Cola 1.5L`; but it can also claim a product range more consistent: for example, it would like to offer `Lipton 2L`, nevertheless, increase cannot be realized product by product, but by subset of products and the final offer should be `Coca-Cola 1.5L` + `Lipton 2L` + `Orangina 1.5L` + `Schweppes 1.5L`.

The proposed approach includes prior knowledge on the hierarchical organization of products by family and constraints on levels of assortment for each family. It proposes to calculate the ideal assortment from the overall point of view of store' managers. The ideal assortment thus appears as a combinatorial optimization problem that can be solved thanks to approximate reasoning based on the products' hierarchy of abstraction. The main difficulty of the optimization problem remains the estimation of the expected benefits associated to any increase of the product range in a given family. This estimate is based on the accounting results of similar stores. The definition of the *similarity* between two stores is then crucial. It is based on the prior knowledge on the hierarchical organization of products that allows approximate reasoning and constitutes the major contribution of this paper.

2 Modelling the Ideal Assortment as a Combinatorial Optimization Problem

Let Ω be the department store.

F_i is the i^{th} family of products, *i.e.* a set of products that are related to a same use category or consumption unit (*e.g.*, `soft-drinks`, `household electrical products`, etc.).

Recursively, any family F_i is a specialization of a super family: *e.g.*, $\texttt{Coca-cola} \in F_{Soda} \subset F_{SoftDrinks} \subset F_{Drinks}$.

Products can thus be organized within a taxonomic partial order defining an abstraction hierarchy (Fig. 1). Products are the most specific classes of this partial order, the leaves of the taxonomy.

Let us distinguish the particular case of families of products, *i.e.* the families the lower in the hierarchy, the less abstracted ones because their descendants are concrete products (direct parents of products). For each of these families of products F_i, a product range or level of assortment $s(F_i)$ is defined: for each family of products, the department store may choose the wideness of $s(F_i)$ in a finite set of opportunities imposed by the direction of the stores' network. Formally, for each family, a hierarchy of subsets of products $s^{k_i}(F_i) = 1..n$ in the sense of the inclusion relationship (*i.e.* $s^{k_i}(F_i) \subset s^{k_i+1}(F_i)$) is defined and the department store can only choose among the subsets $s^{k_i}(F_i)$ as product range for F_i (*e.g.* imagine the minimal product range of the `Soda` family would be `Coca-Cola 1.5L`, the second one `Coca-Cola 1.5L` + `Lipton 2L` + `Orangina 1.5L` + `Schweppes 1.5L`, and so on). Thus, $s(F_i)$ can only be a subset of products that belongs to this finite set of product ranges $s^{k_i}(F_i) = 1..n$ defined a priori by retailer. The size of $s(F_i)$ is then the level k_i of assortment such that $s(F_i) = s^{k_i}(F_i)$. In practice, k_i is a natural number that may vary from 1 to 9. $k_i = 1$ when the product range for the family of products F_i is minimal and $k_i = 9$ when it is maximal.

Therefore, we can write: $\Omega \triangleq \bigcup_{i=1}^{n} s^{k_i}(F_i)$. An expected turnover $p(s^{k_i}(F_i))$ and a storage cost $c(s^{k_i}(F_i))$ can be associated to each $s^{k_i}(F_i)$. $c(s^{k_i}(F_i))$ represents the storage cost the department store allocates for the family F_i.

For any super family in the hierarchical organization of products, its expected turnover and its storage capacity are simply computed recursively as the sum of the expected turnovers and storage capacities of the product it covers.

Designing the assortment of a department store then consists in choosing the rank k_i for each family of products (see Fig. 1). Obviously, the higher $p(\Omega) \triangleq \bigcup_{i=1}^{n} p(s^{k_i}(F_i))$, the better the assortment of Ω. Nevertheless, without further constraints, $p(\Omega)$ should be necessarily maximal when $k_i = 9 \ \forall \ i = 1..n$. In practice, $\sum_{i=1}^{n} c(s^{k_i}(F_i))$ is generally far below $\sum_{i=1}^{n} c(s^{9}(F_i))$ for obvious storage or cost constraints $\mathcal{C}$. Let us consider $\mathcal{I}$ a subset of families. It can be necessary to model constraints related to this super family. For example:
$c(s^{k_{s.drinks}}(F_{s.drinks})) + c(s^{k_{beers}}(F_{beers})) + c(s^{k_{waters}}(F_{waters})) \leq \mathcal{C}_{\mathcal{I}=Beverages}$

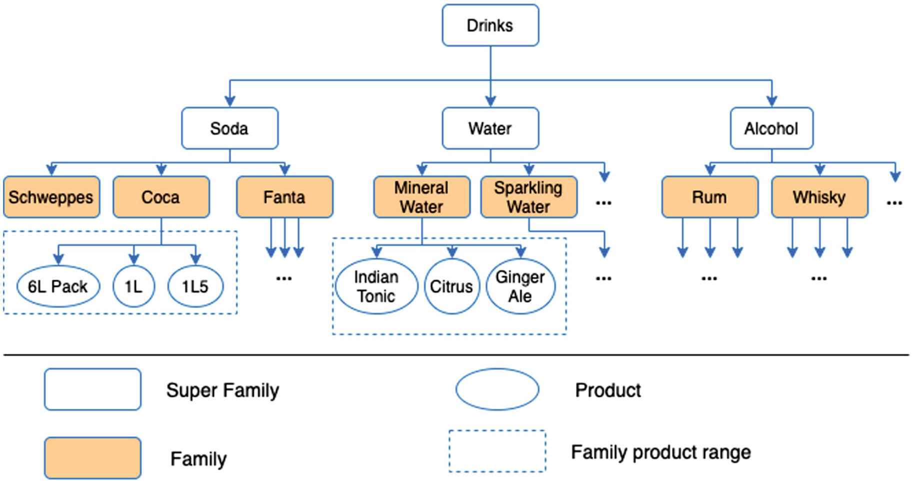


Fig. 1. Products organization and department store assortment as the union of families' product ranges

means that the storage capacity (or the cash flow) related to `Beverages` (superfamily $F_{\mathcal{I}}$) is limited to $\mathcal{C}_{\mathcal{I}}$. A lower $c_{\mathcal{I}}$ bound can also be introduced: in our example, $c_{\mathcal{I}}$ represents the minimal level of investment for the superfamily family `Drinks`. For any superfamily, such local constraints can be added to the optimization problem.

$$\left|\begin{array}{l}
\text{Arg} \max_{k_i, i=1..n} \sum_{i=1}^{n} p(s^{k_i}(F_i)) \\
\text{ Under:} \\
\sum_{i=1}^{n} c(s^{k_i}(F_i)) \leq \mathcal{C} - \textit{global constraint} \\
\text{For some } \mathcal{I} \text{ in } 2^{\{1..n\}}, c_{\mathcal{I}} \leq \sum_{i=1}^{|\mathcal{I}|} c(s^{k_i}(F_i)) \leq \mathcal{C}_{\mathcal{I}} - \textit{global constraint}
\end{array}\right.$$

This combinatorial optimization problem is known as the knapsack problem with mono dimensional constraints and bounded natural number variables.

3 Estimate of the Expected Turnover in the Knapsack Problem

Let consider that one of the assortments to be assessed in the optimization problem includes the increase of the product range of the product family F_i: $s^{k_i}(F_i)$ is upgraded as $s^{k_i+1}(F_i)$. The storage cost (or purchase price) $c(s^{k_i+1}(F_i))$ can be easily completed by the store because it is a basic notion in the retail segment. It is thus easy to inform this point in the optimization problem. On the other hand, it is thornier to estimate $p(s^{k_i+1}(F_i))$ that is however essential to assess the expected performance of the new product range. When the level of assortment of the store is k_i, it is easy to fill in its turnover $p(s^{k_i}(F_i))$ in the knapsack problem but $p(s^{k_i+1}(F_i))$ cannot directly assessed.

We have to design an estimator of $p(s^{k_i+1}(F_i))$. It can only be estimated from other reference measurements encountered in other similar stores. The basic idea is that the more similar these “reference” stores are to the store of concern, the more reliable the estimation. The most difficult problem is to define what “reference” means. Intuitively, the “reference” stores are departments that are “close” to the department store of concern and offer $s^{k_i+1}(F_i)$ to their customers. $p(s^{k_i^{\Omega}+1}(F_i))$ can then be computed for example as the weighted mean or the max of the $p(s^{k_i^{\Omega'}+1}(F_i))$ values, where *Ω*’ are the reference stores neighbours of *Ω*. For sake of simplicity, the neighborhood is restricted to the nearest reference store in our experiments. The next issue is now to define what “close to” means.

This concept of distance between any two department store is the crucial issue. Roughly speaking, *Ω* should be similar to *Ω*’ when the turnovers of *Ω* and *Ω*’ are distributed in the same way over the hierarchical organization of products. It implies they have approximately the same types of customers.

Intuitively, the distance between any two stores should be based on a classical metrics space where the n dimensions would correspond to all the products that are proposed by the department store of a given chain; the value of each coordinate would be the turnover of the product for example, and would be null if the department store does not propose this product. Because some hypermarkets offer up to 100,000 products, the clustering process on such a metrics space would be based on a sparse matrix and then suffer from the space dimension. Furthermore, such a distance would not capture the hierarchical organization of products in the concept of similarity. Indeed, let’s go back to the hierarchical organization of products in families. We can note that a `fruits` and `vegetables` specialist department store is obviously closer to a large grocery store than to a hardware store because the first two are `food` superstores whereas the last one is a speciality store: the first two propose the same super family F_{Food}. This intuitive similarity cannot be assessed with classical distances. The hierarchical products organization in families of products and super families is prior knowledge to be considered when assessing how similar two departments stores are. It is necessary to introduce more appropriate measures that take advantage of this organization to assess the similarity of any two departments store. This notion of similarity measures is detailed in Sect. 4.

In previous sections, we have introduced the levels of assortment $s^{k_i}(F_i), k = 1..n$ for any product family F_i. Note that the increase from $s^{k_i}(F_i)$ to $s^{k_i+1}(F_i)$ must generate an improved turnover for the product family F_i to be worthwhile. By contrast, it requires a higher storage cost $c(s^{k_i+1}(F_i))$ than $c(s^{k_i}(F_i))$. Therefore, the storage cost of at least one product family $F_{j,j \neq i}$ must be reduced to keep the overall storage cost of the department store constant. Then, the reduced turnover $p(s^{k_{j-1}}(F_j))$ of the family F_j has yet to be estimated to complete the optimization problem. However, this estimation can easily be processed. Indeed, because $s^{k_{j-1}}(F_j) \subset s^{k_j}(F_j)$, $p(s^{k_{j-1}}(F_j))$ can simply be deduced from $p(s^{k_j}(F_j))$: it is the sum of the turnovers of all products that belong to $s^{k_j}(F_j) \cap s^{k_{j-1}}(F_j)$. There is clearly an assumption behind this estimation: the disappearance of a product will not change drastically the turnover of other products of the same family. At this stage, for any store Ω, $\forall\ (k_1, k_2, \ldots, k_n) \in [1..9]^n$, we can estimate any $p(s^{k_{i+1}}(F_i))$ as the corresponding turnover of the closest reference store to Ω.

Because designing the assortment of a department store consists in choosing the rank k_i for each family of products, we could now naively enumerate and evaluate any potential assortment in $[1..9]^n$ to select the best one that will be the solution of the optimization problem.

4 Taxonomy and Abstraction Reasoning

The similarity measure that meets our expectations relies on the taxonomical structure that organises products and product families in the department store since Ω should be similar to Ω' when the turnovers of Ω and Ω' are distributed in the same way over the hierarchical organization of products. Generally, in the literature, the elements of the taxonomical structure are named concepts (or classes). A taxonomical structure defines a partial order of the key concepts of a domain by generalizing and specializing relationships between concepts (*e.g.* `Soft drinks` generalizes `Soda` that in turn generalizes `Coca` or `Schweppes`). Taxonomies give access to consensual abstraction of concepts with hierarchical relationships, *e.g.* `Vegetables` defines a class or concept that includes `beans`, `leeks`, `carrots` and so on, that are more specific concepts. Taxonomies are central components of a large variety of applications that rely on computer-processable domain expert knowledge, *e.g.* medical information and clinical decision support systems [1]. They are largely used in Artificial Intelligence systems, Information Retrieval, Computational Linguistics... [2].

In our study, using products taxonomy allows synthetizing and comparing the sales of department store through abstraction reasoning. In retail world, product taxonomy can be achieved by different means. Retailers or other experts can build this commodity structure. Most approaches usually introduce the Stock Keeping Unit (SKU) per item [3] or product categories (*e.g.* `Meat`, `Vegetables`, `Drinks`, etc.). Some researchers adopt the cross-category level indicated by domain experts and/or marketers [4].

More formally, we consider a concept taxonomy $\mathrm{T} = (\preceq, \mathrm{C})$ where (C) stands for the set of concepts (*i.e.* class of products in our case) and $(\preceq)$ the partial ordering. We denote $A(c) = \{x \in \mathrm{C}/c \preceq x\}$ and $D(c) = \{x \in \mathrm{C}/x \preceq c\}$ respectively the ancestors and descendants of the concept $c \in \mathrm{C}$. The root is the unique concept without ancestors (except itself) ($A(root) = \{root\}$) and a concept without descendant (except itself) is denoted a leaf (in our case a leaf is a product) and $D(leaf) = \{leaf\}$. We also denote *leaves-c* the set of leaves (*i.e.* products in our study) that are included in the concept (or class) c, *i.e.*, *leaves-c*$= D(c) \cap$ *leaves*.

4.1 Informativeness Based on Taxonomy

An important aspect of taxonomies is that they give the opportunity to analyse intrinsic and contextual properties of concepts. Indeed, by analysing their topologies and additional information about concept usage, several authors have proposed models, which take advantage of taxonomies in order to estimate the Information Content (IC) of concepts [5]. IC models are designed to mimic human, generally consensual and intuitive, appreciation of concept informativeness. As an example, most people will agree that the concept `Cucumber` is more informative than the concept `Vegetables` in the sense that knowing the fact that a customer buys `Cucumber` is more informative than knowing that he buys `Vegetables`. Indeed, various taxonomy-driven analyses, such as computing the similarity of concepts, extensively depend on accurate IC computational models. Initially, Semantic Similarity Measure (SSM) were designed in an “ad-hoc” manner for few specific domains [6]. Research have been done in order to get a theoretical unifying framework of SSMs and to be able to compare them [1,7].

More formally, we denote I the set of instances, and $I^*(c) \subseteq I$ the instances that are explicitly associated to the concept c. We consider that no annotation associated to an instance can be inferred, *i.e.*, $\forall\ c, c' \in\ C$, with $c \preceq c', I^*(c) \cap I^*(c') = \emptyset$. We denote $I(c) = I$ the instances that are associated to the concept c considering the transitivity of the taxonomic relationship and concept partial ordering $\preceq$, *e.g.* $I(\texttt{Vegetables}) \subseteq I(\texttt{Food})$. We therefore obtain $\forall\ c \in\ C, |I(c)| = \sum\limits_{x\in\ D(c)} |I^*(x)|$.

In our approach, we use sales receipt to count the instances of concept: obviously, only products appear on sales receipt, and then only instances of products can occur in practice. The information is only carried by the leaves of the taxonomy (products in our case), $\forall c \notin$ *leaves*, $|I^*(c)| = 0$.

Due to the transitivity of the taxonomic relationship the instances of a concept $c \in\ C$ are also instances of any concept subsuming c, *i.e.*, `Vegetables` $\preceq$ `Food` $\Rightarrow I(\texttt{Vegetables}) \subseteq (\texttt{Food})$. This central notion is generally used to discuss the specificity of a concept, *i.e.* how restrictive a concept is with regard to I. The more restrictive a concept, the more specific it is considered to be. In the literature, the specificity of a concept is also regarded as the Information Content (IC). In this paper we will refer to the notion of IC defined through a function $IC : C \longrightarrow \mathbb{R}^+$. In accordance to knowledge modelling constraints, any IC function must monotonically decrease from the leaves to the root of the taxonomy such as $c \preceq c' \Rightarrow IC(c) \geq IC(c')$.

In this paper, extrinsic evidence has been used to estimate concept informativeness (*i.e.* that can be found outside the taxonomy). This is an extrinsic approach, based on Shannon’s Information Theory and proposes to assess the informativeness of a concept by analysing a collection of items. Originally defined by Resnik [5], the IC of a concept c is defined to be inversely proportional to

$pro(c)$, the probability that c occurs in a collection. Considering that evidence of concept usage can be obtained by studying a collection of items (here, products) associated to concepts, the probability that an instance of I belongs to $I(c)$ can be defined such as $pro : 2^c \rightarrow [0,1]$ with $pro(c) = |I(c)|/|I|$. The informativeness of a concept is next assessed by defining: $IC(c) = -log(pro(c))$.

We will then use extrinsic IC in our proposal to capture concept usage in our specific application context. Let us note T the taxonomy of products, F the set of families (or classes). The leaves of T are the products (*e.g.*, `Coca-cola 1.5L`). Classes that directly subsume leaves are product families (*e.g.* `Soda`) with which assortment levels are associated ($(k_1, k_2, \ldots, k_n) \in [1..9]^n$); other classes are super family of products (*e.g.*, `Soft-drinks`). Let us derive these notions in our modeling. The above "collection of items" corresponds to the products that a network of department store within a same chain sails. $\forall x \in leaves(T), p^{\Omega}(x)$ (*i.e.* x is a product) is the turnover related to the product x in the store Ω. We define the probability mass *pro* as:

$$\left|\begin{array}{l} \forall x \in leaves(T), |I^*(x)| \triangleq |I(x)| = \sum_{\Omega} p^{\Omega}(x) \text{ then} \\ pro(x) = \dfrac{\sum_{\Omega} p^{\Omega}(x)}{\sum_x \sum_{\Omega} p^{\Omega}(x)} \text{ and } IC(x) = -log(pro(x)) \\ \forall\ f \notin leaves, |I^*(f)| = 0, |I(f)| = \sum_{x \in leaves\text{-}f} |I(x)|, \\ pro(f) = \dfrac{\sum_{x \in leaves\text{-}f} \sum_{\Omega} p^{\Omega}(x)}{\sum_x \sum_{\Omega} p^{\Omega}(x)} \text{ and } IC(f) = -log(pro(f)) \end{array}\right.$$

4.2 Similarity Measures Based on Taxonomy

After the informativeness of a concept is computed, we can now explain how to compute the similarity of any two concepts using concepts' informativeness. We recall some famous Semantic Similarity Measure (SSM) based on the informativeness of concepts and usually used in Information Retrieval. One common SSM is based on the Most Informative Common Ancestor (MICA) also named the Nearest Common Ancestor (NCA). For example, in Fig. 1, the MICA of `Coca-cola 1.5L` and `Schweppes 1.5L` is `Soda` while the MICA of `Soda` and `Water` is `Drinks` (the root in Fig. 1). Resnik [5] is the first to implicitly define the MICA: this is the concept that subsumes two concepts c_1 and c_2 that has the higher IC (*i.e.*, the most specific ancestor):

$$sim_{\text{Resnik}}(c_1, c_2) = IC(MICA(c_1, c_2))$$

Such SSMs allow comparing any two concepts. However, as stores are associated to subsets of concepts, we still have to introduce group similarities to compare sub-sets of concepts. Indirect SSMs have been proposed [8,9]. The Best Match Average (BMA) [8] is a composite average between two sets of concepts, here A and B:

$$sim_{\text{BMA}}(A, B) = \frac{1}{2|B|} \sum_{c \in B} sim_m(c, A) + \frac{1}{2|A|} \sum_{c \in A} sim_m(c, B)$$

where $sim_m(c, X) = \max_{c' \in X} sim(c, c')$ and $sim(c, c')$ is any IC-based pairwise SSM. It is thus the average of all maximum similarities of concepts in A regarding B and vice-versa. This is the most common group similarity. See [8,9] for a complete review.

Pairwise and groupwise SSMs allow comparing any two subsets of concepts (products in our case) when a taxonomical structure defines a partial order of the key concepts of a domain. In our study, they allow to capture the idea that two stores Ω and Ω' are similar when their turnovers are distributed in the same way over the hierarchical organization of products.

5 Illustration and Experiments

This section aims to illustrate the modelling and the data processing chain described in the preceding sections. It is illustrated how designing the ideal assortment in retail thanks to reasoning on an abstraction hierarchy of products, semantic similarity measures and knapsack formalization. The required parameters and variables for this modelling are:

1. A taxonomy of products shared in the store network.
2. A product range (or level of assortment $s^{k_i}(F_i))$ defined for all families F_i of products.
3. A storage cost associated to each product range for each family: for each family, a hierarchy of subsets of products $s^{k_i}(F_i), k_i = 1..n$ is defined, and the higher k_i, the higher the corresponding cost $c(s^{k_i}(F_i))$.
4. For each store, a turnover $p(s^{k_i}(F_i))$ is associated to each product range of each family $s^{k_i}(F_i)$.
5. Storage capacity thresholds are introduced to manage storage constraints (see local constraints in Sect. 2).

Figure 2 illustrates the required data. The example in Fig. 2 takes into account three stores (M1, M2 and M3):

1. We only consider two products' families denoted F for `Fruits` and V for `Vegetables`. There are two product ranges for the `Fruits` family (*i.e.*, k_F is 1 or 2) and three for the `Vegetables` family (*i.e.*, k_V is 1, 2 or 3). We have $S^1(F) \subset S^2(F)$ and $S^1(V) \subset S^2(V) \subset S^3(V)$.
2. Each product range has its own storage cost: $c(S^1(F)) = 346; c(S^2(F)) = 1191; c(S^1(V)) = 204; c(S^2(V)) = 866; c(S^3(V)) = 2400$.
3. From the given product range associated with each store, in this example (k_F, k_V), their turnover can be computed that is $p(S^{k_F}(F)) + p(S^{k_V}(V))$.
4. Each store has the following storage capacity: $SCM_1 = 1670; SCM_2 = 2700; SCM_3 = 5540$ which implies that $c(S^{k_F}(F)) + c(S^{k_V}(V)) \leq SCM$ for each store with given values of a couple of variables (k_F, k_V).

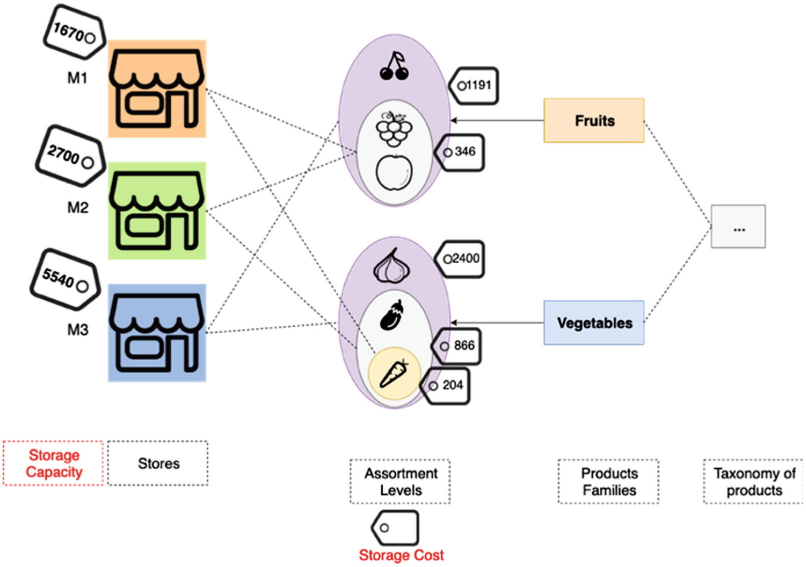
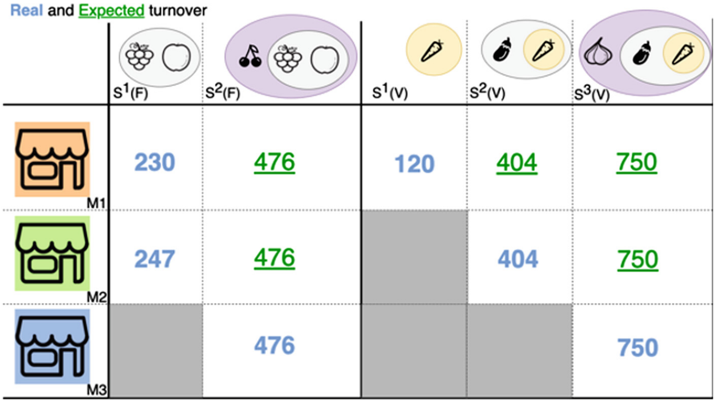


Fig. 2. Required parameters and variables for the knapsack model

Any change in $S^{k_F}(F)$ or $S^{k_V}(V)$ entails turnovers variations. The optimal assortment problem consists in identifying the best couple of values for k_F and k_V. This result is achieved by solving the knapsack problem formalized in this paper. The main difficulty is the assessment of the turnovers when k_F and k_V are changed into $k_F + j$ and $k_V + j$'. An estimation of these turnovers has to be computed in order to evaluate the performance of the candidate values $k_F + j$ and $k_V + j$' in the knapsack problem. As explained above this estimation is based on the turnovers of similar stores that propose $k_F + j$ and $k_V + j$' for families `Fruits` and `Vegetables`. To this end, we apply semantic similarity measures on the product taxonomy to compute a similarity matrix between stores (*cf.* Sect. 4). The unknown turnovers are then assessed from those of the most similar stores. The stores' similarity matrix is based on semantic similarity measures (in this experiment, the Resnik's measure for the semantic similarity measure and the BMA for the groupwise measure using the semantic library tools[1]). Note that, this step allows defining similarities between stores and can be used to define semantic clusters of stores [19]. Once the matrix is defined, it is used to estimate the turnovers of increased candidate product ranges ($k_F + j$ and $k_V + j$') estimated as the corresponding turnovers of the nearest stores that propose $k_F + j$ and $k_V + j$' for F and V. An example of estimation of product range turnovers is proposed in Fig. 3.

[1] https://www.semantic-measures-library.org/sml/index.php?.


Real and Expected turnover

	s1(F)	s2(F)	s1(V)	s2(V)	s3(V)
M1	230	476	120	404	750
M2	247	476		404	750
M3		476			750

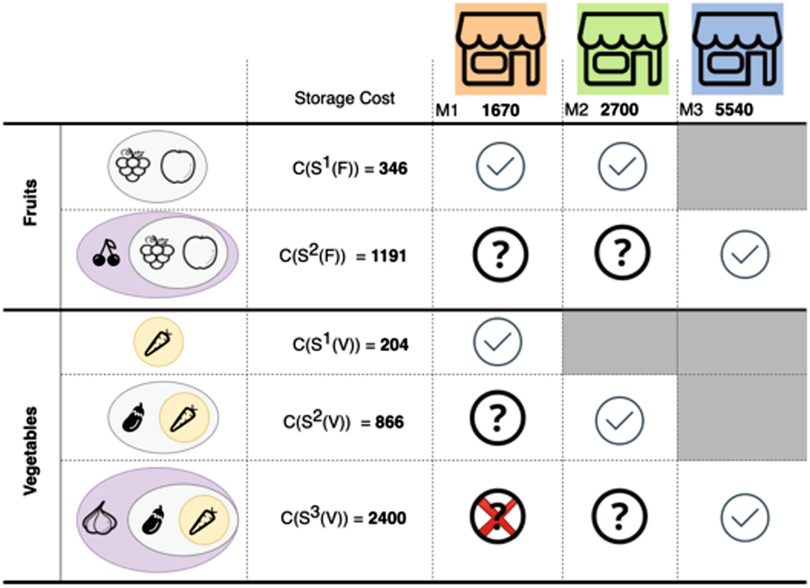
Fig. 3. Estimation of the turnovers of increased product ranges

The last step consists in exploiting the previous estimation in the knapsack problem. As explained above, the knapsack problem aims identifying the ideal product range for each family in order to find: Arg $\max_{k_i, i=1..n} \sum_{i=1}^{n} p(s^{k_i}(F_i))$ while respecting (at least) the overall storage cost constraint $\sum_{i=1}^{n} c(s^{k_i}(F_i)) \leq \mathcal{C}$ defined in Sect. 2. This step involves assessing any combination of $s^{k_i}(F_i)$ for all categories of products F_i. Constraints regarding the storage costs $c(s^{k_i}(F_i))$ can be applied on any category of products which allow reducing complexity of the *knapsack problem* thanks to local reduction of possible solutions (see local constraints in Sect. 2). An illustration on how local constraints reduce the set of solutions is available in the Fig. 4.

For example in Fig. 4, the highest level of assortment for vegetables $s^3(V)$ is greater than the total storage capacity of stores M1 and M2. This information allows eliminating the upgrade $s^3(V)$ for the `Vegetables` family in stores M1 and M2. Finally, in this example three upgrades can be envisaged:

1. Store M1 can improve its `Fruits` assortment from $S^1(F)$ to $S^2(F)$:
$$c(S^2(F)) + c(S^1(V)) \leq \text{SCM}_1$$
2. Store M1 can improve its `Vegetables` assortment from $S^1(V)$ to $S^2(V)$:
$$c(S^1(F)) + c(S^2(V)) \leq \text{SCM}_1$$
3. Store M2 can improve its `Fruits` assortment from $S^1(F)$ to $S^2(F)$:
$$c(S^2(F)) + c(S^2(V)) \leq \text{SCM}_2$$

Then, store M3 owns already all products, so no upgrade is feasible. Due to its storage capacity, store M2 can only improve its `Fruits` assortment. Store



		Storage Cost	M1 1670	M2 2700	M3 5540
Fruits		$C(S^1(F)) = 346$	✓	✓	▨
		$C(S^2(F)) = 1191$	?	?	✓
Vegetables		$C(S^1(V)) = 204$	✓	▨	▨
		$C(S^2(V)) = 866$	?	✓	▨
		$C(S^3(V)) = 2400$	☒	?	✓

Fig. 4. Example of local constraints

M1 is available to improve either its `Fruits` or its `Vegetables` assortments. The Fig. 3 provides the turnovers' estimations for any feasible product range upgrade. The optimal upgrades can now be deduced from it. Therefore, store M1 should upgrade its `Vegetables` assortment from $S^1(V)$ to $S^2(V)$ to improve its turnover. This trivial example allows highlighting how *knapsack problem* can be simplified thanks to local restrictions and taxonomical reasoning.

This example was a mere illustration. The naive optimization of the assortment would consist in trying all possible subsets of products without considering constraints (from stores or from range products). In other words, it requires to try all possible combinations of products whatever their category. In our example, without taxonomy, we should basically reason on the set of products: `apple`, `grapefruit`, `cherry`, `carrot`, `eggplant` and `onion`. With only 6 products, we have 63 possibilities $[2^n - 1]$ which have to be tried for each store. In our toy example, reasoning on the taxonomy of products and managing storage cost constraints significantly reduce the research space. We have shown in other articles referring to the biomedical field the interest of semantic similarities when the dimensions of space are organized by a domain taxonomy [15].

To ensure that this process is scalable with a real dataset from retail, we have built three benchmarks based on the Google Taxonomy[2] that we report in this paper. Experiments have been processed on 1 CPU from an Intel Core I7-2620M 2.7GHz 8Go RAM. We exploited the CPLEX library (IBM CPLEX 1.25) and each benchmark requires less than one second. These benchmarks simply allow

[2] https://www.google.com/basepages/producttype/taxonomy.fr-FR.txt.

us to claim that our complete data processing to compute the ideal assortment of stores of a network can be achieved even for significantly large problems as referred in Table 1. Semantic interpretations of our work are yet to be done and require the intervention of domain experts and evaluations over large periods of time. This assessment is outside the scope of this article and will be carried out as part of the commercial activity of TRF Retail.

Table 1. Benchmarks' details

	Benchmark 1	Benchmark 2	Benchmark 3
Number of stores	15	30	50
Number of levels of range product	4	16	20
Number of families of products	12	80	200
Number of variables	180	2 400	10 000

6 Conclusion

The aim of the paper is to propose a methodology allowing improvement of retailers' assortments. Indeed, the ultimate goal consists in proposing adequate products to stores depending on their specific constraints. To achieve this goal semantic approaches are used not only to improve knowledge on stores but also to make the estimations of the consequences of assortment changes more reliable. As a matter of fact, the proposed approach includes prior knowledge from the taxonomy of products used to formalize the ideal assortment problem into a knapsack problem. The estimation of the expected benefits associated to changes in the product range of products' families is based on results of similar stores. Those similarities are identified by means of semantic similarity measures we previously studied in the field of biomedical information retrieval. The use of semantic approaches brings more appropriate results to retailers because it includes part of their knowledge on the organization of products sold.

The management of the products' taxonomy notably reduces the search space for the knapsack problem. It also allows defining an appropriate similarity matrix between the stores of a network that takes into account the way the turnovers of the stores are distributed. It implies they have approximately the same types of customers. The definition of this similarity is crucial for the estimation of turnovers required in the knapsack problem. This process should be computed repetitively to allow continuous improvement which is a key factor in retail sector. Actually, we are working on the integration of more sophisticated constraints in our optimization problem in order to capture more complex behaviors of retailers.

References

1. Harispe, S., Sanchez, D., Ranwez, S., Janaqi, S., Montmain, J.: A framework for unifying ontology-based semantic similarity measures: a study in the biomedical domain. J. Biomed. Inform. **48**, 38–53 (2014)
2. Harispe, S., Imoussaten, A., Trousset, F., Montmain, J.: On the consideration of a bring-to-mind model for computing the information content of concepts defined into ontologies, pp. 1–8 (2015)
3. Kim, H.K., Kim, J.K., Chen, Q.Y.: A product network analysis for extending the market basket analysis. Expert Syst. Appl. **39**(8), 7403–7410 (2012)
4. lbadvi, A., Shahbazi, M.: A hybrid recommendation technique based on product category attributes. Expert Syst. Appl. **36**(9), 11480–11488 (2009)
5. Resnik, P.: Using information content to evaluate semantic similarity in a taxonomy. In: Proceedings of IJCAI-95, pp. 448–453 (1995)
6. Sanchez, D., Batet, M.: Semantic similarity estimation in the biomedical domain: an ontology-based information-theoretic perspective. J. Biomed. Inform. **44**(5), 749–759 (2011)
7. Janaqi, S., Harispe, S., Ranwez, S., Montmain, J.: Robust selection of domain-specific semantic similarity measures from uncertain expertise. In: Laurent, A., Strauss, O., Bouchon-Meunier, B., Yager, R.R. (eds.) IPMU 2014. CCIS, vol. 444, pp. 1–10. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-08852-5_1
8. Schlicker, A., Domingues, F.S., Rahnenfhrer, J., Lengauer, T.: A new measure for functional similarity of gene products based on gene ontology. BMC Bioinform. **7**, 302 (2006). https://doi.org/10.1186/1471-2105-7-302
9. Pesquita, C., Faria, D., Bastos, H., Falcao, A., Couto, F.: Evaluating go-based semantic similarity measures. In Proceedings of the 10th Annual Bio-Ontologies Meeting, vol. 37, p. 38 (2007)
10. Yucel, E., Karaesmen, F., Salman, F.S., Türkay, M.: Optimizing product assortment under customer-driven demand substitution. Eur. J. Oper. Res. **199**(3), 759–768 (2009)
11. Kok, A.G., Fisher, M.L., Vaidyanathan, R.: Assortment planning: review of literature and industry practice. In: Retail Supply Chain Management, pp. 1-46 (2006)
12. Agrawal, N., Smith, S.A.: Optimal retail assortments for substitutable items purchased in sets. Naval Res. Logist. **50**(7), 793–822 (2003)
13. Alptekinoglu, A.: Mass customization vs. mass production: variety and price competition. Manuf. Serv. Oper. Manag. **6**(1), 98–103 (2004)
14. Boatwright, P., Nunes, J.C.: Reducing assortment: an attribute-based approach. J. Mark. **65**(3), 50–63 (2001)
15. Poncelet, J., Jean, P.A., Trousset, F., Montmain, J.: Impact des mesures de similarité sémantique dans un algorithme de partitionnement : d'un cas biomédical à la détection de comportements de consommation. SFC, pp. 1–6 (2019)
16. Huffman, C., Kahn, B.E.: Variety for sale: mass customization or mass confusion? J. Retail. **74**, 491–513 (1998)
17. Pal, K.: Ontology-based web service architecture for retail supply chain management. Eur. J. Oper. Res. **199**, 759–768 (2009)
18. Netessine, S., Rudi, N.: Centralized and competitive inventory model with demand substitution. Oper. Res. **51**, 329–335 (2003)
19. Poncelet, J., Jean, P.-A., Trousset, F., Montmain, J.: Semantic customers' segmentation. In: El Yacoubi, S., Bagnoli, F., Pacini, G. (eds.) INSCI 2019. LNCS, vol. 11938, pp. 318–325. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-34770-3_26