An Experimental Study on Re-ranking Web Shop Search Results Using Semantic Segmentation of User Profiles

Bram Poppink[a], Flavius Frasincar[a], Tarmo Robal[b,*]

[a] *Erasmus University Rotterdam, PO Box 1738, 3000 DR, Rotterdam, the Netherlands*
[b] *Tallinn University of Technology, Akadeemia tee 15A, 12618, Tallinn, Estonia*

Abstract

Although purchasing online has become increasingly popular, finding suitable products in Web shop environments still requires significant effort from users. This study proposes a method for re-ranking products on a Web shop's search engine to better match individual user preferences using semantic descriptions in user profiles. Through a live experiment, we demonstrate that the proposed re-ranking method incorporating segment-based customer preferences increases the commercial efficiency of Web shop's search engine, increases the average rank of product clicks and the add-to-basket rate, while decreases the click-through rate. Interestingly and contrary to what holds for Web search, the empirical evidence collected in this study indicates that the average rank of clicks used to measure the quality of Web search may not be a reliable indicator for the product search result ranking quality for product search engines in Web shop environments.

Keywords: user profiling, preference ranking, semantic description, user segmentation, shopping behavior, product search, query result re-ranking, e-commerce, online shopping

---

1. Introduction

In the last decade, online shopping has rapidly gained popularity. For example, the number of Web shops in the Netherlands six-folded between 2007 and 2015 from approximately $5,000$ to $30,000$ according to the Dutch National Bureau for statistics CBS[1]. Only in the United States, the spending on e-commerce was estimated to achieve 414 billion dollars in 2018, being 11% of the estimated total retail sales of 2018 [36]. The popularity of Web shops comes from their ability

*Corresponding author
Email addresses: `bram_poppink@hotmail.com` (Bram Poppink), `frasincar@ese.eur.nl` (Flavius Frasincar), `tarmo.robal@ttu.ee` (Tarmo Robal)

[1] `https://www.cbs.nl`

to provide a wide assortment of goods compared to traditional stores, which have limited retail space setting restrictions on goods on display. Furthermore, most Web shops offer home delivery, even cross-border, which makes them a convenient way to purchase products. Moreover, to better answer customers' needs, one can adapt the content to user preferences [8] and also decide on the timing of these changes [24].

However, the ability to provide a wide variety of goods in Web shops has introduced the problem of information overload [9, 52] into the online retail sector [2, 38, 41], as customers now have a lot more options to choose from. To cope with the latter and to facilitate the search and purchase of products, most Web shops use on-site search engines – systems of information retrieval (IR) to allow customers to find the information needed by typing in a short description of the desired product. The difficulty of establishing such IR systems lies in extracting users' (in the context of online shopping – customers') needs and determining the relevance of matched information [3], measured either based on the query or the personal interest of the user executing the query. Most on-site search engines are designed such that the information retrieved matches the entered search terms, i.e., the query, as good as possible. In most cases, this provides satisfactory results. Still, such an approach does not take into account the needs of an individual user and thus lacks personalization [11], which would help to find the products that match the query and are therefore the most relevant to a specific customer.

The necessity to personalize search results comes from the fact that search terms are often short and vulnerable to word ambiguity [4, 35]. The latter can be overcome to some extent by word sense disambiguation algorithms [45]. Still, the problem of short queries and lack of proper context remains. Another approach would be to turn to recommender systems and semantic technologies, which have been proven to be useful for, and often used for improving the search [28, 34, 55, 63] – in particular, to apply ontology-based search engine personalization [16, 57, 58] taking into account the past behaviour of a user to determine the context in which the user is looking for information. Personalized rankings could force products, which align the most with the personal preferences of a user, to the top of a search result lister page – thus making them immediately available for the customer. In the state where search queries are short and little is known about the user context applying segment-based search could increase personal relevancy of the highest ranked results, and thus improve the quality of the search result product lister page.

The objectives of this research are twofold and tackle two main research questions:

RQ1: *Does re-ranking of the products on a search result product lister page using segment-based preferences improve the quality of the search result lister page?* In Web search, this quality improvement is indicated by a decrease in the average ranking of product clicks from a search result lister page. We hypothesise that as is common for Web search personalization, the inclusion of segment-based preferences for product search will lead to a decrease in the average ranking of product clicks from a search result lister page.

RQ2: *What is the effect of the proposed method of segment-based approach on the efficiency of the search engine for products in Web shops?* We hypothesise that including customer preferences on their interest segment level delivers product lister pages with more relevant content causing exploration of products also on lower levels of the lister page (exploration of products with higher ranking), and addresses the information overload, which leads to a possible increase in commercial efficiency.

To answer the two latter research questions, we experiment with re-ranking the search results based on segment-based preferences for De Bijenkorf's Web shop search engine and analyze the empirical Web shop data to see the effects of this approach to user profiling for Web shops. De Bijenkorf – a Dutch retailer with multiple affiliate stores located in the Netherlands selling luxury retail products – started selling products online in 2006. On the Web site of the online store of De Bijenkorf an internal search engine can be used to find information or products. Typically, product search result lister pages consist of more than hundreds of products a user needs to go through to find a suitable one. For instance, a search on the De Bijenkorf's Web shop for "tommy hilfiger" would result in more than 1,500 items available in the returned result set, spanning over 32 product lister pages, 48 products on each – a dreadful information overload to handle for customers who seek for a particular product while specifying only a few search details. The search engine of De Bijenkorf ranks products by their popularity, newness, and availability. By considering customer personal preferences and including the latter in product ranking method, the on-site search engine would be able to help the user in finding the products that match the query and are the most relevant to the specific customer. Our proposed re-ranking method is targeting this personal relevancy and a decrease in the required user search effort in finding a suitable product.

Although the experiments were carried out on De Bijenkorf's Web shop search engine, there are no limitations on using the method on another Web shop platform using different measures or sorting rules for product ranking.

The novelty of this study is the approach we take on user profiling through Web shop product segmentation and using the product taxonomy as the profile schema. Although segmentation is not new, the approach we take here based on the Web shop taxonomy of product categories is new, and we show that the segmentation signal brings added value to the signals (popularity, newness, and availability) currently used by De Bijenkorf from the existing literature, and thereby improves the search quality. The proposed methodology consists of four major steps: first, we construct user profiles where we advantage of the existing product taxonomy; second, we apply dimensionality reduction, followed by k-means clustering; and last, we re-rank search results based on segment average preferences. Our empirical study shows that incorporating segment-based preferences results in an increasing average rank of product clicks (opposite to what holds to Web search), and from a commercial perspective in an increase in the probability to add a product into a basket in an online Web shop.

The article makes the following contributions. First, it proposes a novel approach for user profiling by employing the Web shop product taxonomy, shown to raise customer interest in products presented on the lister page and increase the commercial efficiency of Web shop search results (lower click-through rate and higher add-to-basket rate). Second, the experiments indicate that contrary to Web search engine personalization, we find an increase in the average rank for product clicks due to specific factors that are relevant in an e-commerce setup (e.g., cost, brand, and quality). To our knowledge, this is one of the first works to explore the benefits of user segmentation on the product search results of a Web shop.

The rest of the paper is structured as follows. Section 2 provides an overview of related works, while Section 3 addresses the proposed method for segment-based search results. In Section 4 the experiment with the Web shop of De Bijenkorf and its results are discussed. Last, Section 5 provides conclusions and ideas for further research.

2. Related Work

On one hand, the emergence of online retail has enabled merchants to provide customers with a large variety of brands and products with almost no restrictions. On the other hand, this has caused users of Web shops to face a problem of how to efficiently find the required or desired product. Li [38], Lucian [41], and Aljukhadar et al. [2] studied consumer information overload in online shopping environments. The study of Li [38] revealed that customers tend to perceive a higher level of information overload when the number of product alternatives increases, and yet interestingly the customers do not feel overloaded with information when a greater number of product attributes is presented since these are used as cues to make decisions on purchasing a product. Lucian [41] on the other hand measured the relationship between information overload and consumer satisfaction, finding that information overload causes consumers to feel confused at the time of purchase but paradoxically also the satisfaction will be greater in a digital commerce environment. Aljukhadar et al. [2] investigated how information overload affects customers' decision strategies in e-store environments, linking it to provided recommendations. Their study found that higher levels of information overload make consumers increasingly trust and follow the recommendations the system proposes, and customers let it to influence their decision-making. This is also in line with the findings of Joachims et al. [32] for Web search. Specifically, customers treat the recommendations as signals that other choices in the product assortment might not be optimal. The study by Aljukhadar et al. also indicated that consumers start to experience information overload when there exist 18 or more alternatives to choose from. The authors concluded that product recommendations mitigate the negative effects of information overload. In summary, even though the problem of information overload is somewhat alleviated with the exploitation of search engines, in a typical scenario these still return a large amount of irrelevant information users still have to go through [30]. Personalised search results and recommender systems are seen as a solution to the information overload problem [2, 51]. Clearly, there exists a necessity for personalizing and re-ranking product search results in Web shop environments to facilitate finding suitable products efficiently and to tackle the information overload problem from the perspective of product search and match.

Personalizing Web search results and manipulating the ranking of search results to better fit an individual user has been a research ground for many studies, either targeting fairer ranking to improve topical coverage [14], re-ranking scientific search results by the use of scientometric data [26] or vector norms for page ranking [19], focusing on using click-through data with preference mining and ranking optimization for the task [31, 47], or establishing ontology-based profiles [6, 16, 44, 57, 58] to alter the ranking of a search engine and provide Web search personalization. Still, all these studies have focused on Web search engines, and not on on-site search engines in online shopping sites for product search – the main focus of our study in this paper.

Little attention has been paid to product search engines, while a majority of literature focuses on generic Web search engines. Most product search engines are based on adaptations of theoretical models devised for information retrieval with a focus on optimizing search ranking algorithms for Web documents with relevance as the main measure and products sold online being treated similar to documents [37, 62]. Yet, these two types of search engines serve quite different tasks. While Web search aims to retrieve the most relevant documents, the task of product search is to optimize towards relevance and revenue at the same time. In addition, the decision mechanism behind product search is different and in contrast to Web search consists of two stages – first, performing the search and comparing the products on a search result page, and second, deciding whether to purchase a product on the product description page [62]. While using product search, customers are not only searching for something relevant (as in the case of Web search) but also trying to find the best deal to satisfy their specific needs defined by multiple criteria. Thereby, the use of product search engines involves a more complex process of deciding which product to buy, where additional information (e.g., price, rating, popularity) is also important. Due to the latter, the direct application of traditional Web search approaches and ranking in the context of Web shops is not easy, as also noted by Wu et al. [62].

Web search engines usually rank-order the results and present them with the best match (highest-ranked) on top of the page. Numerous eye tracking studies on users' Web search behaviour have been carried out [20, 21, 23, 32, 40], finding that users rarely look at results at lower end of the Search Engine Results Page (SERP) due to a high confidence in the search engine result ranking [21], and often tend to choose the first few results on the top of the page while ignoring the rest [32]. The eye tracking study on the effect of target rank on Web search by Guan and Cutrell [21] indicated that users were less successful in finding target results when targets were displayed at lower positions in the returned results list and people only go through the results on the top of the list. This also conforms to the results of Pan et al. [49] that users tend to first skim through SERPs and look at the first few search results. Goldberg et al. [20] find that users search is biased towards horizontal search as opposed to vertical. Joachims et al. [32] on the other hand found while studying click-through data with eye tracking that on average relative preferences derived from clicks are reasonably accurate. Moreover, their study revealed that the informed decisions users make while clicking are biased in at least two ways: first, by trusting items ranked higher by the system (trust bias), and second, the decisions made are affected by the quality of other ranked items around (quality bias), and thereby clicks need to be interpreted relative to the order of presentation. The latter findings form a basis for interpreting the results of our proposed product re-ranking method in a Web shop environment.

In order to facilitate search and customize search results for the needs of a particular individual, user preferences are commonly captured as user profiles. Different approaches have been taken to describe user preferences, starting with search histories [11]. Gauch et al. [17] outline three major representation types of user profiles: a set of weighted keywords, a set of semantic networks, and concept-based profiles, where the nodes represent abstract topics considered interesting to the user. Sanchez and Bellogin [55] established user profiles based on interaction sequences and extended the longest common subsequence (LCS) as a similarity metric to the recommendation context. In [53] Salehi and Kamalabadi clustered learners using a k-means algorithm to construct user profiles based on multidimensional attributes of the learning materials, while Zhao et al. [63] extended user preferences with a detected knowledge gap, and Formoso et al. [13] explored expansion methods of the user profile in the context of a new user or new item in the system.

In [16] Gauch et al. used browsing history to construct user profiles by periodically extracting Web pages visited by a user. For this, they constructed the ontology of concepts as a composition of concepts in the subject hierarchies from multiple sources (e.g., Yahoo!, Magellan, Lycos, and the Open Directory Project). The concepts into which the most visited Web pages were classified represented those concepts of greatest user interest. With this approach, a moderate 8% improvement in precision for the top 20 results was shown. Speretta and Gauch [58] showed that personalizing and re-ranking results improved Google's rank over 33%. For their work, Speretta and Gauch used the search history from 6 volunteers over a time span of half a year on Google's search engine to construct profiles (based on search queries or snippets on the search engine result page) using the three-level deep concept hierarchy of the Open Directory Project. Similarly, Sieg et al. [57] used browsing history to construct user profiles using the concepts and relations in the Open Directory Project as a reference ontology to personalize Web search results, and Garcia-Sanchez et al. [15] to build an interest ontology for semantic ads recommender system for users of social networking sites. Calegari and Pasi [6] on the other hand applied the YAGO general purpose ontology for generating user profiles to deliver user topical interests. An algorithm of spreading activation along concepts of interest and concepts related was utilized, gaining significant improvement in recall and precision for search results. These studies [16, 57, 58] can be seen as the basis for ontology-based search personalization. Ontology-based user profiles are utilized in multiple applications such as for personalization of search systems for sustainable urban freight transport [5], personalized food and nutrition recommendations [33], personalized recommendations in the field of tourism and leisure activities [43], and recommending personalized advertisements [15]. Yet, there is a lack of work on personalized search for on-site search engines, in particular in the context of online Web shops.

For e-commerce usually a logical product hierarchy (i.e., product taxonomy) already exists, and thus there is no imminent need to construct a special reference ontology in parallel to product classification. Kouki et al. [34] exploited such product taxonomy to deliver product collection recommendations using both co-purchase information and domain knowledge for the task. Ziegler et al. [64] exploited bulk taxonomic information designed for exact product classification for user profile construction, and to infer profile similarities between users to discover like-minded peers. They claim that the use of taxonomies is a powerful background knowledge, which they use as a cornerstone of their recommender system, where users are not represented by their respecting product-rating vectors but by vectors of interest scores for topics (and not particular product instances) in the product taxonomy. Hung [25] took advantage of such a natural product hierarchy on an e-commerce Web site and used it as the reference ontology for the creation of user profiles for a personalized recommendation system. A modified product taxonomy with a root node, a two-level deep category structure, and a third and fourth level brand structure was established to identify user preferences in terms of product categories and brands. Grouping like-minded individuals and creating peer groups takes it further towards collaborative recommendation where the shared interest can be used to enrich recommendations for an individual. In [61] Weng and Liu took a collaborative recommendation approach and used a k-means algorithm to find customers with similar interests based on purchase transaction records. The potential of their approach relies on being able to recommend products which are relatively different from past purchases of an individual but are being purchased by other customers belonging to the same cluster.

The goal of segmentation, i.e., clustering, in marketing research is to create groups of individuals such that the similarity of characteristics of individuals in the same group is maximized, or the similarity of characteristics of individuals in different groups is minimized. Segmentation can be separated into model-based segmentation, and segmentation based on machine learning algorithms allowing to cluster unstructured data into meaningful groups without assuming an underlying model. A type of machine learning clustering technique is partitional clustering, where data is divided into a predefined number of groups based on properly defined optimization criteria [48]. An intuitive partitional clustering technique with convenient properties is found in k-means clustering, first introduced in [12] and with alternative approaches proposed over time [22, 39, 42]. The k-means clustering is an easy-to-implement algorithm to group "like-minded" customers into clusters and is applicable to very large data sets.

In our research we are interested in creating personalized weighted concept hierarchies, similar to the ones created for ontology-based search personalization by Gauch et al. [16] and Sieg et al. [57], utilizing a modified version of product hierarchy of a Web shop as a reference ontology inspired by the work of Hung [25]. Carried by the idea of collaborative recommendation [61], we then focus on creating segments of user profiles [12] to serve segment-based search result rankings for the on-site Web shop search engine in order to improve the quality of product search results. As current literature focuses on Web search personalization and lacks discussion over using an existing product hierarchy for Web shop search engine result personalization and improvement, we also address this gap in product search with our work, binding ontology-based personalization to collaborative recommendation over e-commerce product hierarchies and customer segmentation.

3. Method for Segment-Based Search Results

For e-commerce, it is common to present products in a logical hierarchy, which can be used to browse through the assortment to the desired product, or information [46]. These hierarchies can serve as product taxonomies [25]. The use of product taxonomies or ontologies to represent user profiles is not new, and has been exploited in many research works [25, 34, 64]. Herein,

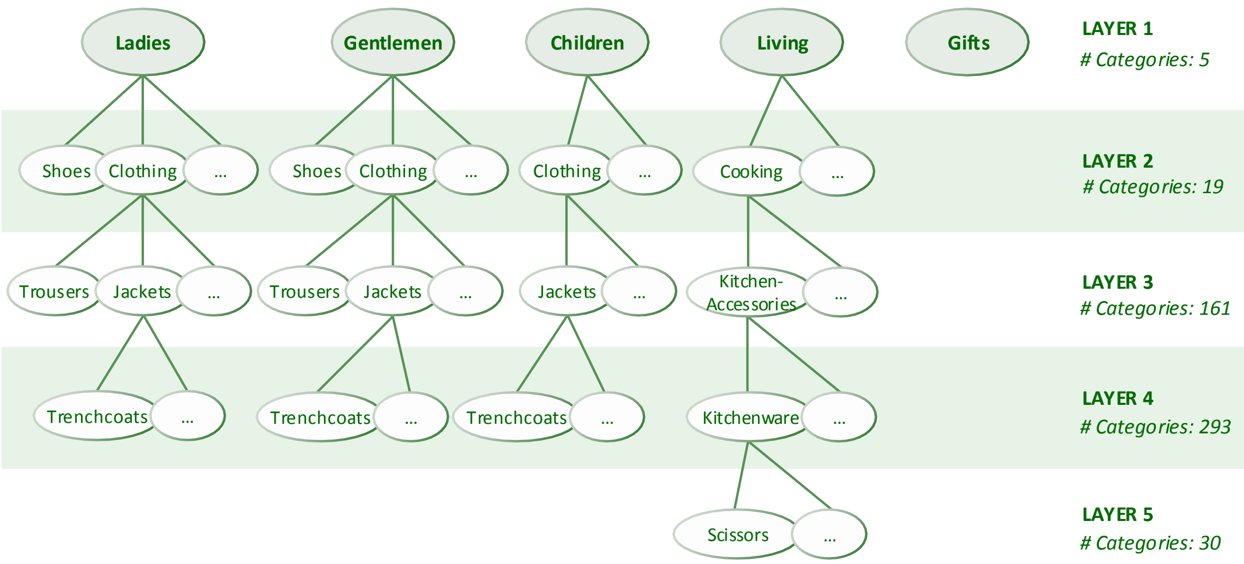

Figure 1: Visualization of the product hierarchy available on the De Bijenkorf's Web site.

we take a similar approach and use the product hierarchy to define a modified (for our purpose) product taxonomy, which will be utilized as a reference ontology for the creation of weighted concept hierarchies per individual, i.e., customer profiles, as is common in the context of ontology-based Web search personalization [16, 57, 58]. The use of a product taxonomy for user profile construction allows us to infer similarities between users even if they have no specific products in common. In particular, for our research we use the product hierarchy of the luxury department store De Bijenkorf consisting of five category layers which can be interpreted as an ontology. Figure 1 provides a simplified overview of this product hierarchy (not all concepts on layers 2, 3, 4, and 5 are shown) due to a large number of categories (e.g., *Layer* 3 contains 161 categories), and the fact that most products are categorized up to *Layer* 3 only, and the category *Gifts* is not categorized at all besides the first layer. We take advantage of this existing product hierarchy setup for the Web shop of De Bijenkorf and exploit a modified version of this product taxonomy to form the basis of a user profile in our method. Like in [64], this allows us to find like-minded customers.

The method we propose for serving segment-based search results (an overview is depicted in Figure 2) consists of four main steps: (i) construction of user profiles, (ii) dimensionality reduction using principal component analysis (PCA), (iii) k-means clustering, and (iv) re-ranking of search results. First, user profiles are constructed based on product detail page views differentiated per user. For the classification of all products a product taxonomy is utilized, such that the page views

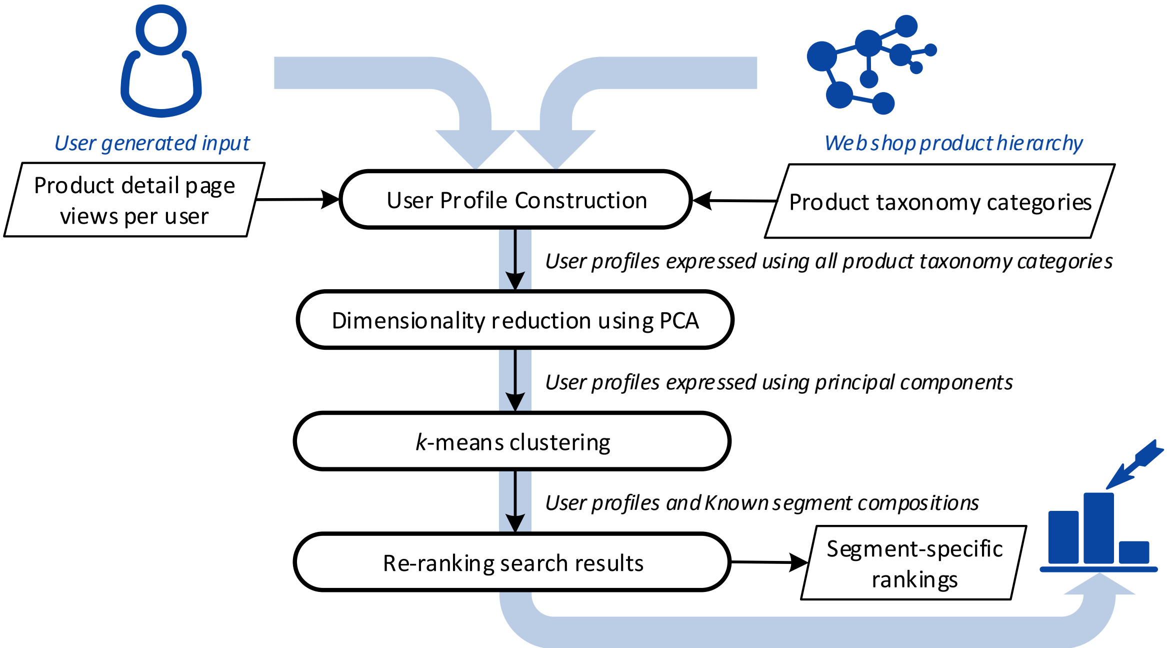

Figure 2: Overview of the proposed method for segment-based search for Web shop.

can be used as implicit feedback on what type of products the user prefers, and thus user preference profiles can be constructed. Next, dimensionality reduction using linear PCA is applied to prepare for k-means clustering. In the third step, all users are assigned to a cluster based on the user profile, resulting in segmentation of customers into groups of "like-minded" individuals, i.e., users with similar product preferences. As the last step, re-ranking based on segment average preferences is applied to determine the segment-specific product preferences and to re-rank products on the search result product lister page accordingly without affecting any other ranking mechanism used by the Web shop nor using any specifics of the considered Web shop, making the approach generic. Let us now view each of these four steps in more detail, starting with modifications to the product taxonomy and construction of the user preference profiles.

3.1. From Product Hierarchy to User Profiles

To establish customer profiles we used the product hierarchy of the De Bijenkorf (sketched in Figure 1). In particular, we modified this hierarchy and use it as a reference ontology for the creation of weighted concept hierarchies per individual. The purpose of the modification is to refine the hierarchy and remove some deficiencies in the Web shop product taxonomy. Namely, the

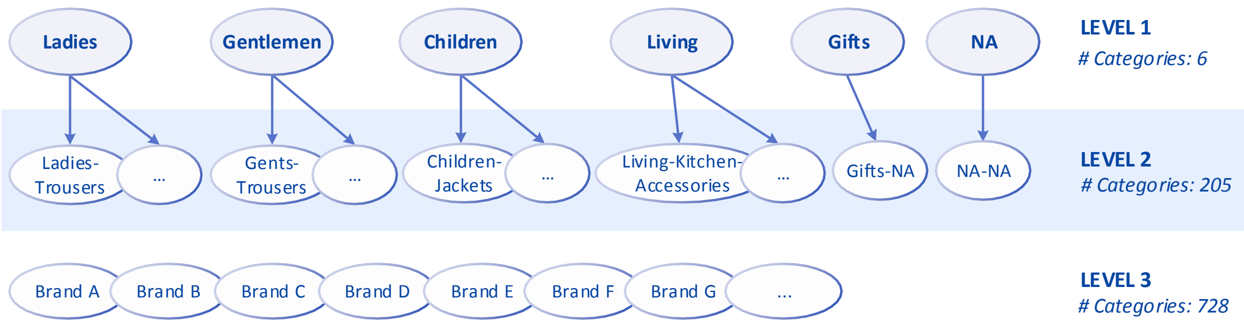

Figure 3: Simplified visualization of the modified product taxonomy.

existing hierarchy (depicted in Figure 1) has five layers with a lot of missing values in *Layer* 4 and *Layer* 5, since not all products are categorized for those layers and these layers are mainly used for the products of the *Living Department* category. Also, there is no layer in the hierarchy concerning the brand of a product, which is fundamental for constructing customer profiles. Furthermore, *Layer* 2 and *Layer* 3 both contain similar product type information where logically *Layer* 3 is more detailed than *Layer* 2; whereas including only *Layer* 3 would be sufficient to determine user product type preference and to simplify the hierarchy. Figure 3 presents the modified product taxonomy. With the modification, *Level* 1 is incorporated as the first layer to account for a broad split of products into meaningful departments (6 categories), while the third layer of the initial hierarchy is incorporated in the modified taxonomy as the second level (*Level* 2, 205 categories) to account for product type preference, and a new *Level* 3 with all available brands (728 different categories) is introduced to account for brand preference in the user profile construction. Not to loose any information for products that were not classified into a category, a new grouping as *Not Available (NA)* was introduced for all missing values (e.g., *Ladies-NA*), guaranteeing that even for unclassified products an approximate interest score can be determined. As the brand of the product is always available, no missing values occur on *Level* 3. The construction of modified product taxonomy is necessary for the construction of user profiles expressed through the product taxonomy categories. The design approach of refining the existing product hierarchy into a reference ontology we introduced here is general enough to be re-used as a starting point for any Web shop product taxonomy refinement in order to allow user profile creation based on product hierarchies.

In our approach user (i.e., Web shop customer) profiles are established as instances of the product taxonomy in which weighted interest scores (WIS) are registered for every concept in the taxonomy through all the three levels in the user profile. The user profiles are initialized similarly for every user by assigning an interest score (IS) of 0 to every concept when no information about the interests and preferences are available. When a user interacts with the Web shop and exposes interest in a particular product by viewing the product detail page, information about user preferences can be deducted and registered in the user profile by increasing the value of IS for the concepts the product belongs to by 1 on every level (the sum of page views on each level is the same). This approach ensures the changing user preferences are accounted at all times, and preferences of new users are obtained as early as possible. The concepts which obtain the highest number of page views represent the concepts of most interest to the user. Although here we only focus on existing customers and our proposed method does not directly address the problem of cold users (new customers), the latter could be potentially overcome by recommending the most popular items or using active learning to gauge the interest of new users [18].

Formally, if each node $j \in 1, ..., J$ in the personalized concept hierarchy for individual $i \in 1, ..., N$ is described as a pair $\Big(C_j, IS_i(C_j)\Big)$ consisting of the concept C_j and the corresponding interest score of individual i to concept j, i.e., $IS_i(C_j)$, the weighted interest score for a concept in user profile is expressed as:

$$WIS_i(C_j) = \frac{IS_i(C_j)}{\sum_{j \in J_l} IS_i(C_j)}, \forall j \in J_l, \tag{1}$$

where J_l is the set of concepts belonging to level $l = \{1, 2, 3\}$ in the taxonomy. The use of WIS over IS provides in our method an immediate indication of the relative importance of a concept on a particular level (remember, the sum of scores per level equals to 1), and also makes it easier to cluster groups of users with similar relative product interests in the forthcoming steps, where PCA and k-means clustering are applied. Consequently, a profile for user i can be expressed as a vector space $\boldsymbol{u}_i$ of weighted interest scores corresponding to all concepts in the taxonomy:

$$\boldsymbol{u}_i = (WIS_i(C_1), WIS_i(C_2), ..., WIS_i(C_J)). \tag{2}$$

Ultimately, this establishes a matrix of user profiles, where every row represents the weighted concept hierarchy of one user, and thus every column represents one of the concepts in the modified product taxonomy, for which a weighted interest score is determined, per user. An example of user profile construction based on the transformation of product detail page views of a user to a weighted concept hierarchy is provided in Appendix A.

3.2. Segmentation

Next, the constructed user profiles are utilized to cluster customers into segments of like-minded users – groups of customers with similar shopping preferences for products. The decision to use segment-based recommendations has two main advantages. First, it improves performance as calculations can be done on a cluster basis instead of a user basis as is the case in personalization, and second it allows to generalize over similar users, i.e., not only rely on preferences in customer's history but also to recommend products customers with similar preferences were interested in. Segmentation analysis [50] is performed on user profiles consisting of weighted interest scores for the concepts in the modified product taxonomy. Namely, a machine learning clustering algorithm k-means clustering [12], as commonly used and easy to implement partitional clustering algorithm [48], is applied. The advantage of this clustering method is that it optimizes a global objective function and is relatively fast compared to other clustering methods (e.g., hierarchical clustering, density-based clustering, or spectral clustering) [59]. To tackle the drawbacks of the k-means clustering, namely that the number of clusters needs to be specified a priori and k-means clustering is dependent on the Euclidean distance measure which does not perform well for high dimensional datasets, dimensionality reduction is applied beforehand.

Six common approaches exist for determining the number of clusters in a dataset [54] – cross-validation, penalized likelihood estimation, permutation tests, re-sampling, and finding the "knee" of an error curve. The last of this list is the most intuitive solution. The idea is to plot the error of the solution for a range of values for k, i.e., the number of clusters. This solution is known as the L-method [54]. An equivalent and intuitive solution is proposed by Caliński and Harabasz [7]. The idea is similar, however, the measure to be plotted for different values of k is different. The idea is to apply k-means clustering for an arbitrary, yet a reasonable range of values for k, determine the Caliński-Harabasz (CH) index, a cluster validity measure, and set k to be the value for which the highest CH index was found.

The k-means clustering algorithm depends on the use of the Euclidean distance measure. In our case the modified product taxonomy contains 939 concepts, thus the construction of user profiles using these 939 concepts implies that k-means clustering is applied in a 939-dimensional Euclidean space. In the literature [1, 27, 60] it is claimed that the Euclidean distance measure does not perform well in a high-dimensional space. To tackle the problem of the Euclidean distance measure, we use a common practice called dimensionality reduction, i.e., factor analysis techniques, such as principal component analysis (PCA) [27], for dataset pre-processing and dimensionality reduction before applying k-means clustering.

The PCA aims to describe the variables in a dataset as good as possible by a smaller set of orthogonal variables that are a linear combination of the original variables, ensuring that the new dimensions, called principal components, are uncorrelated. The key in dimensionality reduction using PCA is in choosing the smallest number of principal components such that as much variance of the variables in a dataset is explained. It imposes a trade-off between reducing dimensionality and maintaining sufficient variance from the original data accounted for in the transformed data. The most commonly accepted method is plotting the added variance sequentially for the first few principal components (i.e., the first few columns in the data matrix) in a scree plot, which is useful for determining how many principal components need to be kept to capture most of the variability of the data [59]. Searching for the "elbow" or "knee", in a scree plot and choosing all components up to and including the "elbow", will ensure that the principal components adding significant value to the so-called variance accounted will be used. The set of chosen principal components is assumed to be a proper representation of the initial dataset, containing a sufficient amount of the variance present in the initial dataset, but with reduced dimensionality. The newly constructed dataset can subsequently be used as an input for k-means clustering.

To decide the optimal number of principal components, we constructed a scree plot [59] shown in Figure 4, indicating the largest drop in the proportion of variance to occur between the first and second principal components already. This means the user profiles can be the most optimally summarized using two principal components, as the first two principal components preserve most of the variability in our data. Plotting all user profiles based on the first two principal components yields a smoothed scatter plot (Figure 5), with two identifiable high-density groups of users – one on the left side and one on the right side.

We now use the data summarized in the first two principal components from the PCA analysis to determine the number of clusters using the L-method [54]. First, k-means clustering will be applied for a range of values for k. Next, the total within the sum of squares (TWSS) is determined for every solution of k and plotted for the different values of k. To overcome the problem of local optima [22], we take a rather conservative approach and execute the k-means clustering using 25

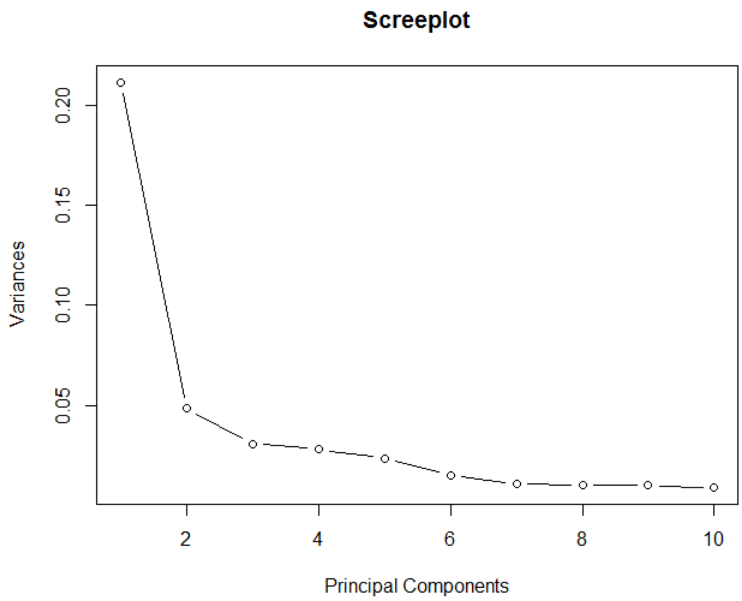

Figure 4: Screeplot for dimensionality reduction for the matrix of user profiles.

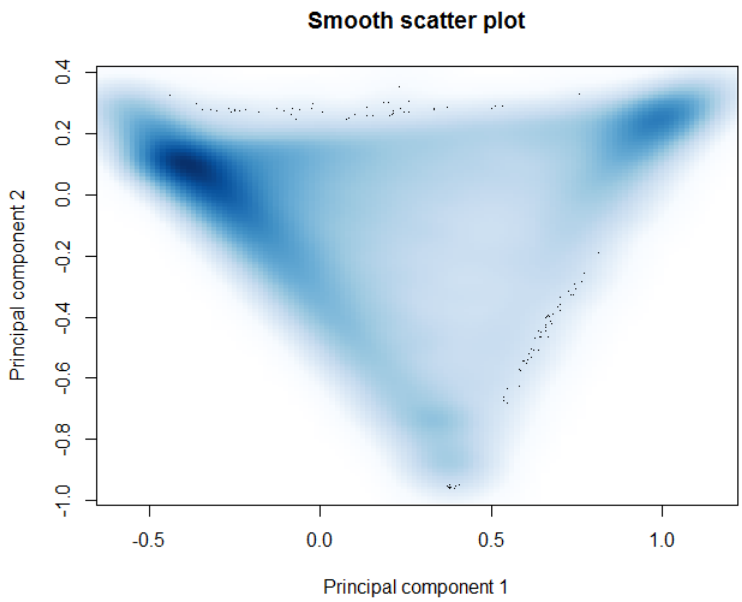

Figure 5: Smooth scatter plot of all user profiles based on the first two principal components.

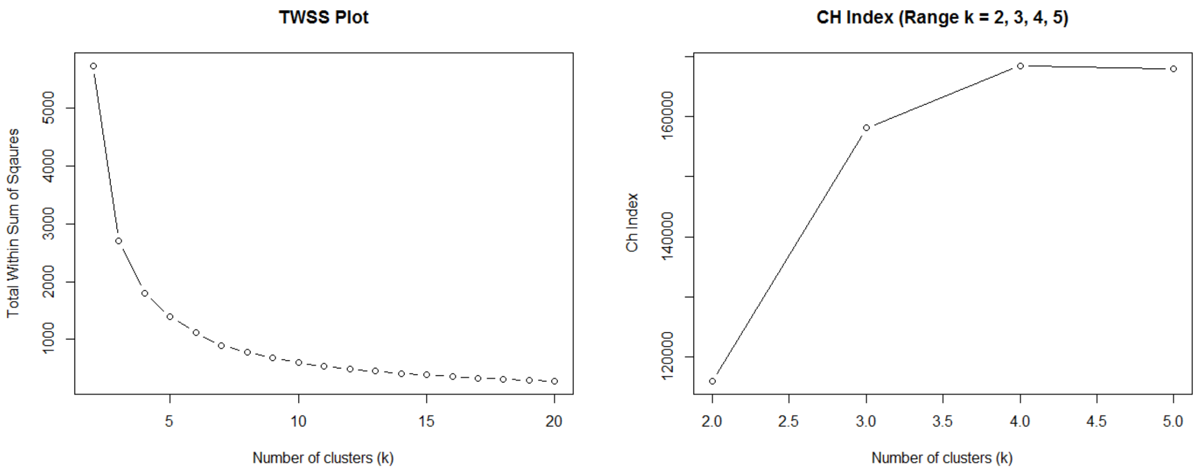

Figure 6: Determining the number of clusters: plot of TWSS values (left); plot of CH index (right).

random starts. Equation 3 provides the formula for TWSS:

$$TWSS = \sum_{s=1}^{k} \sum_{i \in N_s} \sum_{j=1}^{J} (x_{i,j} - c_{s,j})^2 \tag{3}$$

where k is the number of clusters, s is the cluster index, N_s is the set of observations belonging to cluster s, i is the observation index, J is the total number of dimensions used to cluster the observations, and j is a dimension index. The TWSS values for a range of values $k = \{2, 3, ..., 20\}$ are plotted on Figure 6 (left), where one can observe a visible "knee" at $k = 3$, $k = 4$, or $k = 5$, out of which $k = 4$ is the most convincing. To confirm this, the CH index [7] is plotted for the values $k = \{2, 3, 4, 5\}$ (Figure 6 (right)). The highest CH index value is attained at $k = 4$. It appears that for the given set of user profiles the number of clusters is $k = 4$.

Running the clustering algorithm with a setting $k = 4$ using the first two principal components as input data delivers 4 segments such that every user is assigned to one of the clusters. After matching cluster allocations to original user profiles to determine segment average weighted concept hierarchies for centroid calculation, the relative importance of concepts can be found (Table 1).

One can observe (from Table 1) that *Segment* 1 consists of users mainly interested in womenswear, *Segment* 2 of customers mainly interested in menswear, *Segment* 3 of users interested in children clothing and items related to living, and lastly, *Segment* 4 consists of individuals with a significant preference for womenswear but also menswear, living, and uncategorized products. Furthermore, *Gifts* is the category of least interest for all segments. Additionally, *NA* is a substantial

Table 1: The relative importance of the six concepts in level 1.

Level 1	Users	Gifts	Ladies	Gentlemen	Children	Living	NA
Segment 1	30,669	0.000	0.879	0.024	0.011	0.022	0.064
Segment 2	10,840	0.000	0.058	0.841	0.024	0.029	0.047
Segment 3	5,698	0.000	0.175	0.092	0.367	0.291	0.074
Segment 4	15,569	0.000	0.585	0.140	0.070	0.101	0.105

category for all segments which implies that a significant amount of page views were observed for uncategorized products. Detailed descriptions of the segment compositions on *Level 2* and *Level 3* of the modified product taxonomy are presented in Appendix B.

To be able to compare the results of the experiments on the four segments, we define a control group (a fifth group) out of 25% of all individuals randomly selected from the entire dataset so that this group represents individuals from all four different segments. Users in the control group will get served the default (without segmentation) rankings when consulting the search engine.

3.3. Segment-Based Search Result Rankings

Before we continue with the calculation of segment-based product preferences, let us address the meaning of *lower* and *higher* product click ranking. Namely, a *lower* ranking implies that a product is shown *higher* on the lister page. Furthermore, a *decrease* in average ranking, implies that on average products were found *higher* on the search result lister page, and (in the Web search context, e.g, [58]) is interpreted as a measure of the quality of the lister page.

When the k-means clustering is applied, all observations are clustered into groups of observations having similar personalized weighted concept hierarchies, providing clusters of groups of individuals with similar interests and preferences. This allows to calculate the cluster average weighted concept hierarchy through determining average weighted interest scores (WIS) per concept. Thereafter, when the cluster average weighted concept hierarchies are defined, the average relative preference for every product must be determined in order to re-rank the search results. In ontology-based search personalization, the cosine similarity between a vector representation of the user profile and the document in the set of search results is used to measure the preference for the document [57, 58], and the similarity between a profile and document is used as the interest in a certain document which is needed for the re-ranking of the search result set documents. With our method, a product is with certainty assigned to multiple concepts in the product taxonomy for which the relative importance, i.e., segment preference, is known. This allows to avoid cosine similarity, and use a more intuitive and easier to calculate product average WIS measure.

To determine the preference for every product per segment, first the WIS's in the personalized concept hierarchies are averaged per concept over all observations within a cluster. For this, the average interest score per concept is determined for every cluster and the averages are represented in the same vector space as the personalized concept hierarchy. The vector for cluster average weighted concept hierarchy $\bar{\mathbf{c}}_s$, is found as:

$$\bar{\mathbf{c}}_s = \left(\frac{\sum_{i=1}^{N_s} WIS_{i,(s)}(C_1)}{N_s}, \frac{\sum_{i=1}^{N_s} WIS_{i,(s)}(C_2)}{N_s}, ..., \frac{\sum_{i=1}^{N_s} WIS_{i,(s)}(C_J)}{N_s} \right) \tag{4}$$

where $s = 1, ..., k$ is the cluster index, $WIS_{i,(s)}(C_j)$ is the weighted interest of individual i which was assigned to cluster s for concept j, for $i = 1, ..., N_s$, and N_s is the total number of observations assigned to cluster s.

Subsequently, every product q is represented in the same vector space as the cluster average weighted concept hierarchy:

$$\mathbf{p}^q = \left(PS^{(q)}(C_1), PS^{(q)}(C_2), ..., PS^{(q)}(C_J) \right), \tag{5}$$

where $PS^{(q)}(C_j)$ corresponds to the product score of product q for concept j, and every concept to which product q belongs is assigned a product score of 1 (i.e., $PS^{(q)}(C_j) = 1$), and all other concepts a score of 0. Thereafter, the dot product of the product vector $\mathbf{p}^q$ and the cluster average weighted concept hierarchy $\bar{\mathbf{c}}_s$ is determined, yielding the sum of weights for every concept to which the product is assigned. Next, this vector is divided by the number of concepts to which the product was assigned to obtain the average weight of those concepts. This measure is an alternative to the cosine similarity, and more intuitive and easier to calculate. Thus, the relative preference of individuals in cluster s for product q is given by the average product interest score (APIS):

$$APIS_{(s)}^{(q)} = \frac{(\mathbf{p}^q)' \cdot \bar{\mathbf{c}}_s}{\sum_{j=1}^{J} PS^{(q)}(C_j)}, \tag{6}$$

which can be interpreted as the interest in product q for individuals in that cluster. This interest rate represents the segment-specific match rate and is calculated per different segment for all products separately.

The relative importance of products within a segment is used for the re-ranking of search results. At first, the search engine processes the query and determines which products match the query entered. A product only has a match with the query when it possesses all objective product characteristics, e.g., colour, size, etc., used in the query. This results in the search result set, $\mathbf{r} = \{r_1, ..., r_R\}$, where $\mathbf{r} \subset \mathbf{p} = \{p_1, ..., p_P\}$ – a subset of the set of all products. Then the k-means clustering is performed to determine the cluster to which the user (i.e., customer searching for products) belongs to. Since $\mathbf{r}$ is a subset of the set of all products, and the cluster to which the user belongs is known, the segment-based product match rates (given by $APIS$) defined in Equation 6 can be established.

Further, the final match rate for product $q \in \{1, ..., R\}$ and cluster $s \in \{1, ..., k\}$ is calculated as a combination of segment-based search result and general product rankings (default measures used at De Bijenkorf) of product popularity, newness, and availability such as:

$$\begin{aligned} finalMatchRate^{(q)}_{(s)} &= \alpha \cdot segmentMatchRate^{(q)}_{(s)} + \beta \cdot generalPopularity^{(q)} \\ &\quad + \gamma \cdot newness^{(q)} + (1 - \alpha - \beta - \gamma) \cdot availability^{(q)}, \end{aligned} \tag{7}$$

where $\alpha \in (0, 1)$ is a tuning parameter determining the weight assigned to the segment match rate, $\beta \in (0, 1)$ is a parameter defining the weight assigned to the general popularity of a product, $\gamma \in (0, 1)$ defines the weight of the newness of a product, and $1 - \alpha - \beta - \gamma$ the weight of the availability, i.e., stock of a product.

The values of the default measures used to determine the final match rate at De Bijenkorf are constructed as follows. The *general popularity* of a product is the number of page views of a product over all users in the last 7 days and is used by the reasoning that popular products should be shown higher on a search result lister page. *Newness*, expressed by the number of days a product has been available on the Web site, is used to show products that are new in the assortment higher on a lister page (higher value equals lower position on the lister page, because De Bijenkorf is a fashion outlet and newer products are preferred). Including newness tackles the so-called *cold start* problem, when the new product has not gained many page views yet but might be of high interest for customers. Availability incorporates the current stock of a product in the ranking since products that are out of stock should be ranked lower. The value for this measure is given as $availability = \frac{S^{(q)}}{TS^{(q)}}$, where $S^{(q)}$ is the number of available variations (e.g., sizes trough XS to XL) of the product q in stock, and $TS^{(q)}$ is the total number of the product q variations. For example, if a product has sizes: S, M, L, and XL and there are only 2 products of size S and 1 product of size XL available, then the availability $= 2/4$.

As the four different measures in the final match rate formula (Equation 7) are measured on a different scale, the values are normalized to a range between 0 and 1 using the min-max normalization:

$$x' = \frac{x - \min(\mathbf{x})}{\max(\mathbf{x}) - \min(\mathbf{x})} \tag{8}$$

where x is an initial element in the vector corresponding to measure $\mathbf{x}$ and x' is the corresponding scaled value of this element. As *newness* is the only measure for which lower values are favoured over higher values, its scaling must be reversed, thus minimum is replaced by maximum and vice versa. Only when all measures of the final match rate formula are on the same scale, the summation makes sense. Lastly, the final match rate determines the ranking for product r, namely a higher final match rate corresponds to a lower ranking value, i.e., higher location on the search result lister page. The result set r is determined real-time by the search engine based on the entered query, and thus the computation of the final match rates is also determined real-time. The values for all measures influencing the final match rate (segment average interest scores for all products, general popularity, newness, and availability) are updated nightly, remaining the same throughout a period of a day. The pseudo-code given in Algorithm 1 describes the proposed re-ranking process.

4. Experiment and Results

To validate whether segment-based search result rankings improve the search quality, i.e., enhance the efficiency of the search result lister page, experiments were carried out on the Web shop of the luxury department store De Bijenkorf. First, starting from 21 April 2017 for 55 days training data was collected from users who visited the Web shop in at least 10 sessions on distinct days to cover users' long-term preferences. The underlying argument justifying this choice is that page views in one session, or at least in a small time interval, might deviate from users' long-term preferences. Therefore, a minimum of 10 sessions is assumed to rule out the possibility that the product detail page views gathered over the time interval are not aligning with the long-term preferences, i.e., the data represents users' long-term preferences, which are in the focus for this study. This provided a sample of 62,776 individuals who either viewed a product page via the Web shop navigation menu or used the Web shop search engine and viewed a product page through the search

Algorithm 1 Re-ranking algorithm

Input: Datasets of user profiles separated for k clusters and dataset of product score vectors

Output: Relative importance within clusters for all products

$\mathbf{X}_s \in \mathbb{R}^{N_s \times J}$ for all $s = 1, ..., k$; // All personalized weighted concept hierarchies in k different matrices
$\mathbf{P} = \{\mathbf{p}^1, ..., \mathbf{p}^P\} \in \mathbb{R}^{P \times J}$; // Matrix of product score vectors for all P products
for all $s \in \{1, ..., k\}$ **do**
  Calculate $\bar{\mathbf{c}}_s$; // Compute vector of average weighted interest scores (column means)
  for all $q \in \{1, ..., P\}$ **do**
    Calculate $APIS^{(q)}_{(s)}$; // Compute cluster average interest score for all products
  end for
end for
for all New searches **do**
  Search engine determines $\mathbf{r} = \{r_1, ..., r_R\}$; // Set of products that match all characteristics specified in the query
  Determine s, the cluster for current user;
  for all $q \in \{1, ..., R\}$ **do**
    Retrieve $APIS^{(q)}_{(s)}$; // Determine the interest score for product q (element of the result set), for cluster s
    Determine $finalMatchRate^{(q)}_{(s)}$; // Based on segment match rate, popularity of the product, newness of a product, and availability of the product
  end for
  Rank products in the result set $\mathbf{r}$ in descending order of $finalMatchRate$;
end for

result lister page. For these individuals, $662,360$ product clicks from a search result lister page were recorded, generated by $38,829$ users in the sample. It should be noted that all the product detail page views – not only those from a search result lister page but also product page views derived via the Web shop site navigation within the aforementioned time frame – are used to construct user profiles, and this is the data to which we refer as the training data. User profiles were constructed for all $62,776$ users only once and were not updated during the experiment. Firstly, this is due to the fact that the proposed method depends on a segmentation analysis, which would also need to be re-executed every time the profiles are updated, requiring computation time. Secondly, it was assumed that it is highly unlikely for user profiles to change drastically over a short period of time, and thus re-executing the segmentation analysis continuously would be superfluous. We acknowledge that for a longer term user profiles need to be updated over time.

Second, a separate experiment from 25 August 2017 until 4 September 2017 was executed in which every user in the training data was potentially part of, dependent on whether a user visited the Web shop and utilized the search engine over the course of the experiment, or not. The setup of this experiment is discussed in Section 4.1, and the data in Section 4.2.

It is important to highlight that when a user utilizes the search engine of De Bijenkorf's Web shop, the entered query is processed and all the products that match all the characteristics specified in the query are returned. For instance, if a user searches for *yellow sweater*, only products that are both *yellow* and are a *sweater* are included in the result set. Still, product characteristics can be ambiguous, e.g., when a user searches for *polo*, polo shirts are returned but also products from the brands *Marco Polo* and *Polo Ralph Lauren*. During the experiment, the original results on a search lister page will be modified according to the proposed model, and its effect on customers' behaviour is studied. The customers continue to use the Web shop as previously according to the terms and conditions of the Web shop.

It is important to note that for the analysis of the search results, only clicks on products on the *first page* after a search are tracked as search result lister product clicks, and any further pages are interpreted as normal lister pages. The first result page after a search consists of 48 products organized as a grid of 3x16 items presented by text and product images; thus the maximum ranking of a clicked product is equal to 48. Figure 7 presents the product search lister page (the first page after a search presented to a user) model for the De Bijenkorf's Web shop, outlining the potential browser viewport. One can observe that even the first two browser viewports potentially do not enable to follow half of the listing. During the experiment, it was observed that a fourth of product clicks (i.e., clicks performed by a customer to open a product description page from the search result lister page) is lower or equal than rank 4, half of the product clicks are below rank 11, and 75% of clicks are at or below rank 23, with the overall mean rank of clicked products at 14.5 (the first third of the product search lister page). The data about the behaviour of people searching on De Bijenkorf's Web site indicates that most people tend to click on products ranked at the top, yet a significant amount of clicks is also observed even for the lowest ranks on the search result lister page for which the users need to scroll vertically. The clicks performed by users on lister page items with higher ranking located below the initial browser viewport indicate the users' interest (a match for search query) in these items [21, 32, 49].

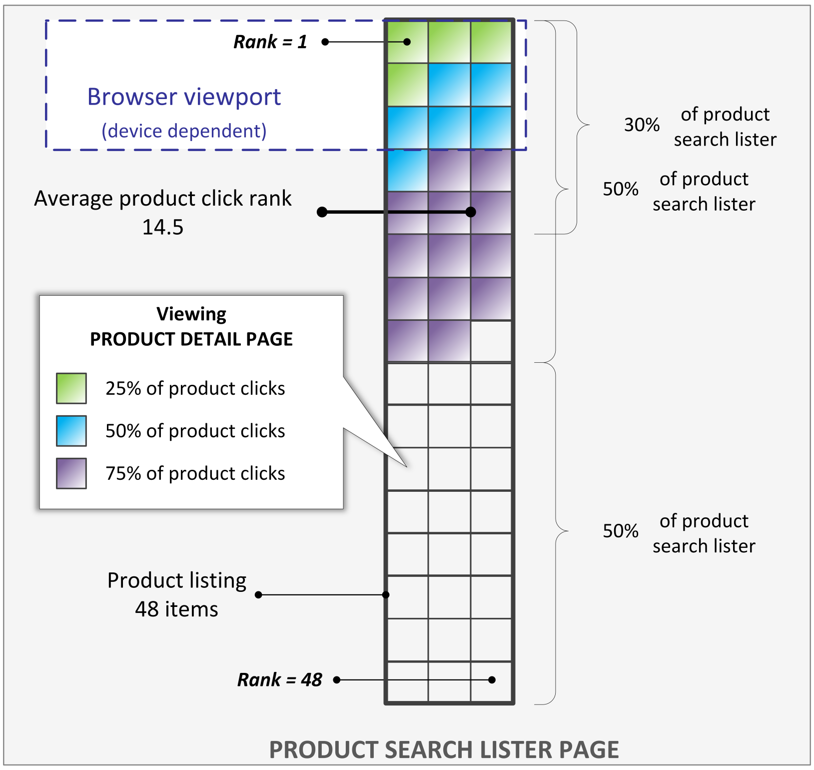

Figure 7: Product search lister page model for the De Bijenkorf's Web site, where search results are arranged as a grid of 3x16 products.

4.1. Experimental Design

To observe an increasing influence of the segment-based match rate on the quality of the search result product lister page the parameters α, β, γ, and $1 - \alpha - \beta - \gamma$ that determine the final match rate in Equation 7 are given multiple values by increasing the influence of parameter α determining the weight of the segment-based match rate while retaining the original proportion of all other parameters, i.e., β, γ, and $1-\alpha-\beta-\gamma$. In all the experiment settings we ensure the ratios of the popularity, newness, and availability weights are kept the same in order to be able to properly measure the effects of the *segmentMatchRate* in various settings. Thus, the parameter α is leading and the other parameter values are distributed equivalently according to their original importance in the initial search engine system (see Table 2 *Control setting*) over the remaining weight to be assigned as follows: $\beta \approx 0.45 * (1 - \alpha)$, $\gamma \approx 0.35 * (1 - \alpha)$, and $1 - \alpha - \beta - \gamma \approx 0.2 * (1 - \alpha)$. Table 2 lists the four different weight parameter settings that are used in the experiments. The final match rate is determined in real-time by the search engine system according to the entered

Table 2: Weight settings for the experiments.

Setting	Time span	α ($segmentMatchRate$)	β ($generalPopularity$)	γ ($newness$)	$1-\alpha-\beta-\gamma$ ($availability$)
Control setting	Continuously	–	0.45	0.35	0.2
Setting 1	Time span 1	0.2	0.36	0.28	0.16
Setting 2	Time span 2	0.5	0.23	0.17	0.10
Setting 3	Time span 3	0.8	0.09	0.07	0.04

query, and the input data, i.e., α, β, γ, and $1-\alpha-\beta-\gamma$, values that are manually specified in the system in advance.

The *Control setting* is the default setting used at De Bijenkorf to rank products in the Web shop advantaging from product popularity, newness and availability, and will be used as the benchmark performance to which the proposed method is compared to. The default setting will be served to a randomly selected subset consisting of 25% of the total sample, denoted by the *Control group*, leaving 75% of the sample for three treatment groups. This provides a situation where the experimental data recorded will be approximately equal for all four settings. The *Control group* will be observed simultaneously with all different settings, while *Setting* 1, *Setting* 2, and *Setting* 3 are observed over three different time slots of the same length. Simultaneous observation of *Settings* 1, 2, and 3 is unfortunately not possible due to technical limitations of De Bijenkorf's Web shop.

The leading statistic for the experiments is similar to the one used in ontology-based search personalization [58], namely the average ranking of the search result clicks. Similarly to ontology-based search personalization, where it was proven that personalized search result rankings result in a decrease in the average ranking of the search result clicks, we assume segment-based search will produce a decrease in the average ranking of all the clicks from a search result lister page onto a product detail page. This decrease implies that the desired information is found faster, namely higher on the page where the products are listed. Yet, this assumption, based on findings for Web search personalization, might not be true and needs to be proven in the context of segment-based Web shop search.

Further, the three different values of α (Table 2) – minor, average or major influence – will be observed together with commercial measures such as click-through rate (CTR) and add-to-basket rate ($AtBR$) to determine the optimal setting of α for the segment-based search, and confirm the efficiency of the proposed approach. The CTR is the number of product clicks relative to the number of searches in session (unique Web site visit):

$$CTR_j = \frac{PC_j}{S_j}, \tag{9}$$

where PC_j is the number of product clicks in session j and S_j denotes the number of searches in session j. The overall CTR for N sessions is the weighted mean CTR_j over all $j = 1, ..., N$, where S_j can be interpreted as the weight assigned to session j, is given by:

$$CTR = \frac{\sum_{j=1}^{N} S_j \cdot CTR_j}{\sum_{j=1}^{N} S_j}. \tag{10}$$

The other commercial measure $AtBR$ is the number of attributable add-to-basket events as the result of a product click from a search result lister page expressed as:

$$AtBR_j = \frac{AtB_j}{PC_j}, \tag{11}$$

where AtB_j is the number of add-to-basket events, attributable to a search result lister, in session j, and PC_j is the number of product clicks from a search result lister page in session j. The overall $AtBR$ as the weighted mean of the $AtBR_j$ over all sessions $j = 1, ..., N$, where PC_j can be interpreted as the weight assigned to the session j, is given by:

$$AtBR = \frac{\sum_{j=1}^{N} PC_j \cdot AtBR_j}{\sum_{j=1}^{N} PC_j}. \tag{12}$$

The fundamental statistic used to analyze the quality of a search product lister page is the average ranking of the product clicks from a search result lister page onto a product detail page. According to the experts at De Bijenkorf, where our data is originating from, customer behaviour can vary substantially day by day due to external factors such as the weather, day of the week, and more importantly internal factors such as promotions, the arrival of new products, or the start or end of a sale period. Our study is conducted outside of any sales period.

To test the validity of the *Control group* (25% of randomly selected individuals from the entire dataset) before the experiments, we looked at the average ranking trend over the course of 55 days (the collected training data) for both the entire sample and the *Control group*. The daily average

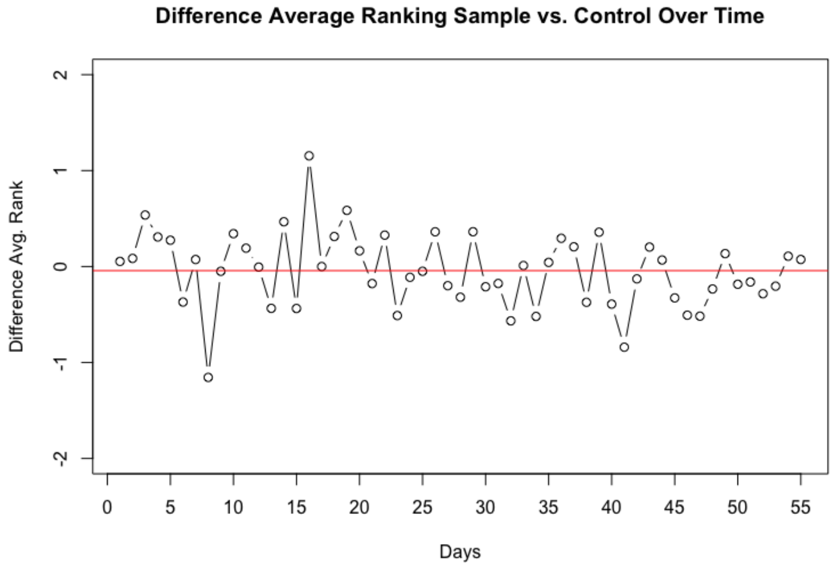

Figure 8: Daily average ranking difference between the entire sample and the control group over time.

ranks of the product clicks for both data sets were in between 13 and 16. The average difference over the studied period is -0.043, plotted as a horizontal line in Figure 8. No consistency in the difference between the sample and the control group is observed, with the difference varying day by day around 0 in the range of -1 and +1. Even though the average rank of the composed control group slightly differs on a daily basis from the entire sample, the difference fluctuates around zero and in the long-term the observed difference is negligibly small (-0.043), it can be concluded that the *Control group* is a trustworthy representation of the sample average ranking.

4.2. Experiment Data

In order to obtain empirical results for all settings of the proposed method, an experiment was executed over the course of ten consecutive days, where data was collected for three different time intervals of exactly 72 hours for the three different weights assigned to α leading the segment-based search result (Table 2).

The experiment started on 25 August 2017 at 13:00 Central European Time (CET) and ended on 04 September 2017 at 13:00 CET. The exact time intervals used for analysis in chronological order are given by 25 August 2017 13:00 until 28 August 2017 13:00 for $\alpha = 0.8$, 28 August 2017 14:00 until 31 August 2017 14:00 for $\alpha = 0.2$, and 1 September 2017 13:00 until 4 September

Table 3: Quantities of observed events and number of users for the entire experiment.

	# Searches	# Product Clicks	# Add to Baskets	# Sessions	# Users
Control	20,324	3,010	313	2,499	1,228
Setting 1	19,435	2,665	248	2,433	1,647
Setting 2	21,401	3,050	405	2,633	1,838
Setting 3	17,804	2,440	284	2,284	1,594
Total	78,964	11,165	1,250	9,849	4,991*

*–distinct users

2017 13:00 for $\alpha = 0.5$. All the three sequential intervals were outside of a sale period, where customer behaviour is known to be different from non-sale periods. The *Control group* is observed continuously and simultaneously with all three different settings. Thus the experimental data will contain approximately 25% *Control group* data, 25% data corresponding to *Setting* 1, 25% data corresponding to *Setting* 2, and 25% data corresponding to *Setting* 3. Still, the data collected per setting will not be exactly uniformly distributed since the data collection depends on how many users per setting utilize the search engine of the Web shop within the different time intervals. Table 3 indicates the distribution of collected data sets over the four settings. Note that the total number of users 4,991 represents the sum of distinct users, and is less than the sum of users over the different settings of α. This difference is caused by users being active in more than one of the three different time intervals corresponding to either $\alpha = 0.2$, 0.5, or 0.8.

4.3. Average Ranking Product Clicks

To observe the effect of the weight assigned to the segment match rate on customer behaviour, we analyze the average ranking of the product clicks from a search lister to a product detail page.

The average rankings for different values of α separated for the treatment group, i.e., segment-based search results, and the *Control group* ($\alpha = 0$) to which the default ranking is served, indicate that increasing the value of α increases the average rank of the product clicks for the treatment group, whereas a downward trend is observed over the three time spans for the *Control group* (as shown in Figure 9). For the treatment group the lowest average rank, given by 13.5, is attained for $\alpha = 0.2$ during time span 1, while the highest average rank, given by 14, is attained for $\alpha = 0.8$.

The daily average rank of the product clicks throughout the data collection period was observed to be ranging between 13 to 16.

Since the changes in average rankings are not tremendous, we proceed to explore whether the difference between the treatment's and control group's average rankings are significantly different from zero. We observe for the percentage difference in mean average ranking of the treatment relative to the control group (given in Figure 10) a decrease of 5.83% for $\alpha = 0.2$, a decrease of 1.16% for $\alpha = 0.5$, while for $\alpha = 0.8$ an increase of 2.89% is found. Overall, for an increasing value of α, an increasing average ranking of the product clicks relative to the control group is observed. In [21] it was shown that users are less successful in finding target results when these are displayed at lower positions in the returned results lister page. From the increasing average ranking of product clicks, we conclude that the inclusion of the segment-based ranking has improved product findability.

Interestingly, the initial assumption that for a personalized (or segment-based) lister page the average rank decreases for higher values of α – proven to be correct in Web search personalization

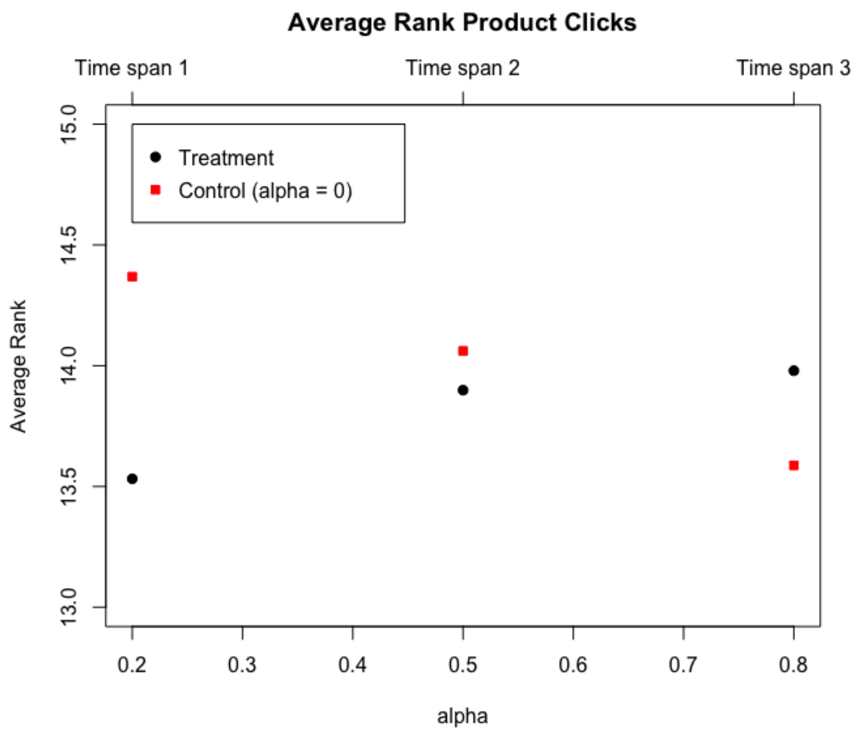

Figure 9: Average ranks of product clicks for the three different settings of α and the control group.

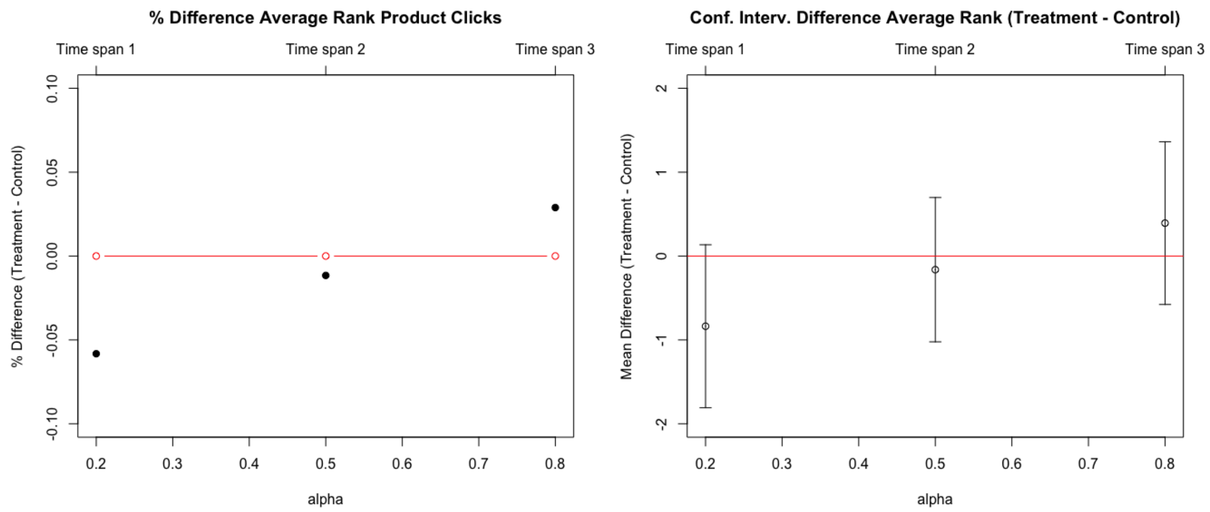

Figure 10: The percentage difference in the treatment's relative to the control group's average rank (left) and the 95% confidence interval of the difference in the average rank of the treatment group minus the average rank of the control group (right) for the three different settings (time span 1-3).

– does not hold in the context of a Web shop product search engine. In fact, our research indicates the opposite – incorporating the segment-based relevance of products in the search result product ranking for users with preferences similar to the segment average preference increases the average rank of the product clicks.

Our research results indicate that in a Web shop setting incorporating the user's personal (product) preferences, by applying segment-based search result product ranking, does not necessarily decrease the average ranking of product clicks. Although it might sound counter-intuitive (since it was proven to be a correct assumption in Web search personalization), one must not forget that there are subtle differences between the stimulus for using a Web shop search engine for finding a desired product and the stimulus for using a Web search engine for retrieving information documents. Namely, when utilizing the Web search engine the user is commonly looking for one piece of information that answers a question or fulfils a need. Contrarily, a Web shop customer uses the on-site product search engine to filter for the most relevant products aligning their needs, but the existence of just one most optimal result, i.e., the product to buy, is more debatable, since the process involves motivational and financial factors that strongly affect decision making [38, 56]. As more relevant products are presented on the lister page, it is easier for users to explore the alternatives and visually compare products listed on the same page, and thereby they are inclined to scroll further down while searching for the desired product. Further, it is perfectly possible that multiple, characteristically different, products satisfy Web shop users' needs, and therefore they are more inclined to compare products on the search result product lister page and are less prone to be directly satisfied with one of the top search results common to users of Web search.

As shown, an improvement of the search result product rankings based on semantic segmentation of user profiles improves the relevance of the top-ranked products and thus enhances the 48 product suggestions on the first lister page after a product search. Based on the results displayed in our research, this apparently results in users (customers of the Web shop using the search engine) scrolling down further on the lister page, probably since more products align with their personal preferences compared to the default non-personalized approach, and thus increasing the product click rankings on average – more products of interest are found lower on the search result lister page, whereas the interest is reflected through opening up product detail pages, indicated as product clicks [32]. In other words, users are not anymore merely satisfied with the topmost search results after a search while ignoring the rest of the product items lower on the page, i.e., products with a higher rank, as common for Web search [21, 32, 49]. On the contrary, when segment-based search results are served by the Web shop, users continue to show interest also for product items lower on the page having higher rank, driving the increase of average product ranking.

Example on Application of the Method

To exemplify the application of semantic segmentation and its effect to search results lister page and product ranking, let us consider a search query "*tommy hilfiger*" executed by the user of the Web shop, and two product search lister pages (Figure 11 and 12) displayed as a result. Figure 11 presents the first 9 results (out of 48) of a product lister page for search "tommy hilfiger" for the *Control group*, representing the default "as-is" case of the search without our method applied on the De Bijenkorf Web shop. The final match rate (Equation 7) for ranking in these results is affected by product popularity, newness, and availability, where $\alpha = 0$, and values for β, γ, $1 - \alpha - \beta - \gamma$ are as presented in Table 2 (Control setting), i.e., $\alpha = 0$, $\beta = 0.45$, $\gamma = 0.35$, and $1 - \alpha - \beta - \gamma = 0.2$. Figure 12 shows the same search results for *Segment* 2 – only menswear is shown, which could be a very beneficial alteration of the rankings for this customer segment. The values of product popularity, newness, and availability come from the De Bijenkorf system, and we can only change their weight through β, γ, and $1 - \alpha - \beta - \gamma$ for the final match rate.

In our method, we create a vector with average interest scores for all categories in our product taxonomy (Equation 4) for each segment. For instance, let us consider two products: *Product 1* as the first product (top-left) on Figure 11 (*Pullover*), which is a product that belongs to *Gentlemen* in Layer 1 of the modified product taxonomy, to *GentlemenSweaters&Cardigans* in Layer 2, and *TommyHilfiger* in Layer 3, and *Product 2* as the second product (from top-left) on Figure 12 (*White Polo*), which is a product that belongs to *Gentlemen* in Layer 1 of the modified product taxonomy, to *GentlemenPolos&Tshirts* in Layer 2, and *TommyHilfiger* in Layer 3. Figure 13 depicts for these two example products their segment-specific average interest scores and product match rates (APIS). These score values are always between 0 and 1, where 0 indicates no interest at all for that category in product taxonomy, and 1 indicates the highest interest.

Next, a vector of length $68,093$ (the number of products on the Web shop of the De Bijenkorf at the time of the experiment) of average product interest scores (APIS) is generated for each segment (Equation 6), indicating the interest in a product for individuals in a cluster (segment). This vector contains a value between 0 and 1 for every product. For instance, the final segment-specific product interest score (APIS) value for *Product 1* for Segment 1 is (Figure 13 bottom-left): $segmentMatchRate = (0.0243 + 0.0013 + 0.0317)/N = 0.0573/3 = 0.0191$, where N is the number of categories the product belongs to. In our example, the products belong to 3 categories.

For the re-ranking of products shown on the lister page as a result of a search, the final match rate is calculated (Equation 7), with values normalized to a range between 0 and 1 using the min-max normalization (Equation 8). This is done for all the measures: segmentMatchRate, generalPopularity, newness (do recall that the scaling here is reversed, i.e., max-min normalization), and availability. Table 4 lists the min-max values for the four different measures used, and their corresponding values for these measures for our example products. For instance, the APIS value for *Product 2* is $segmentMatchRate = 0.3595$, and after the normalisation: $segmentMatchRate = (0.3595 - 0.0060)/(0.3677 - 0.0060) = 0.3535/0.3617 = 0.9773$

Once all the values are normalized, the final match rate can be calculated (Equation 7). For example, the final match for *Product 2* for Segment 2 (with medium effect of semantic segmentation, $\alpha = 0.5$, as given on Figure 11) will be: $finalMatchRate = 0.5 \cdot 0.9773 + 0.23 \cdot 0.075 + 0.17 \cdot 0.9611 + 0.1 \cdot 0.6 = 0.7293$. Table 5 lists the final match rates for *Product 1* and *Product 2* for the default (Control) and segment-based (Segment 2) product listing with normalized input values.

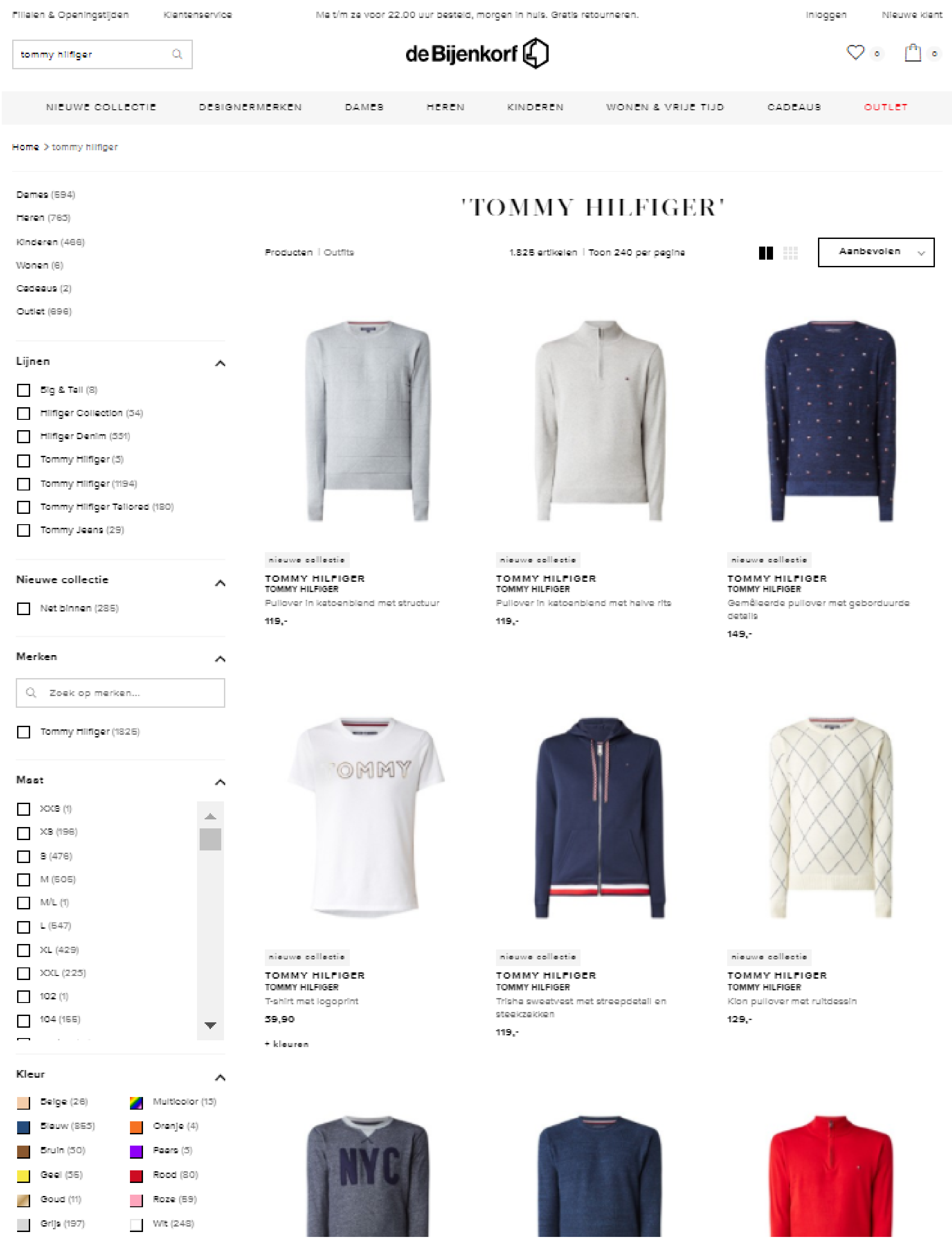

Figure 11: Default rankings of the first 9 products after a search for the brand "tommy hilfiger". The default rankings are served to the control group where the parameter values are given by $\alpha = 0$, $\beta = 0.45$, $\gamma = 0.35$, and $1 - \alpha - \beta - \gamma = 0.2$.

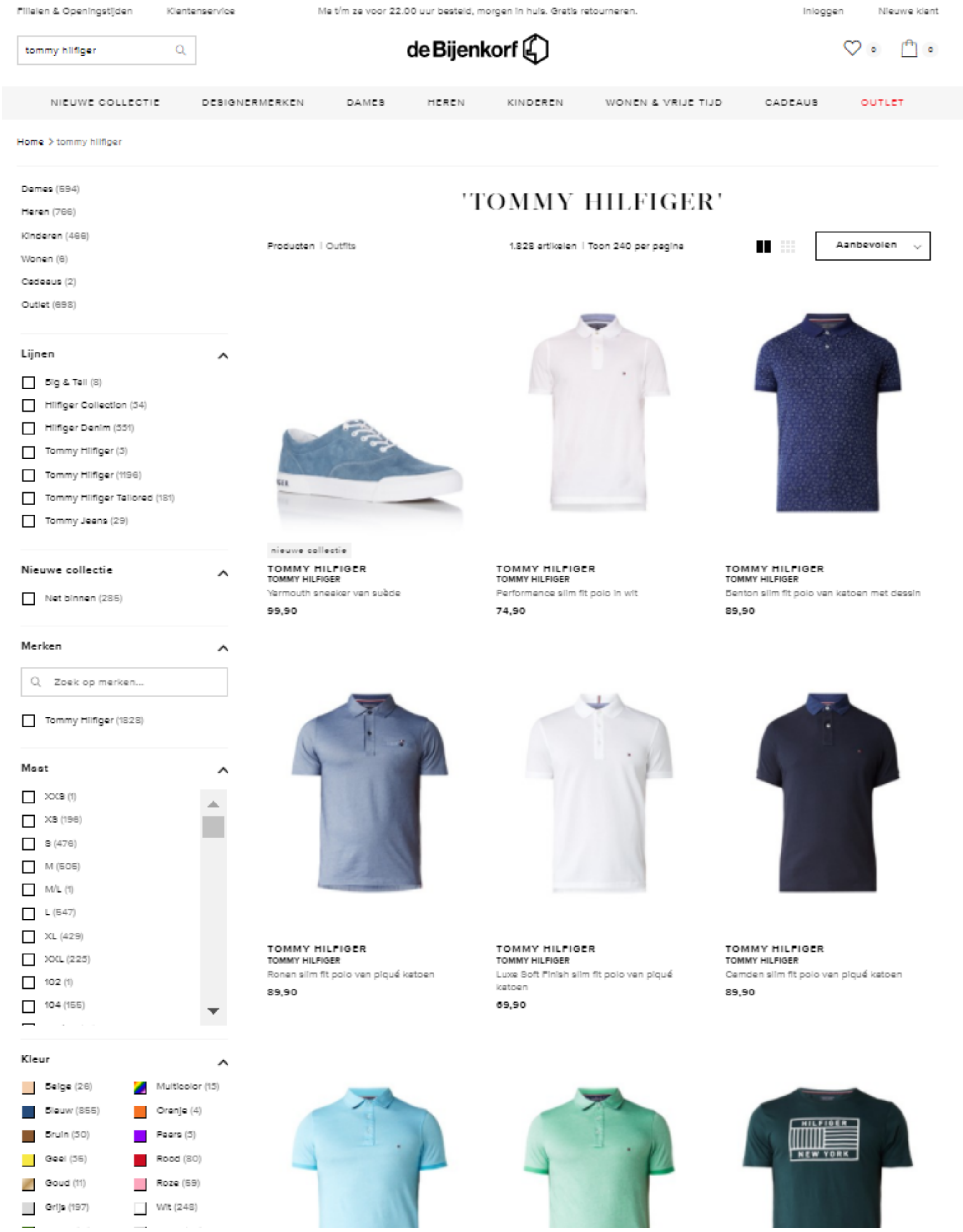

Figure 12: Rankings of the first 9 products after a search for the brand "tommy hilfiger" for Segment 2. The rankings are served to Segment 2, where the parameter values are set to $\alpha = 0.5$, $\beta = 0.23$, $\gamma = 0.17$, and $1 - \alpha - \beta - \gamma = 0.1$.

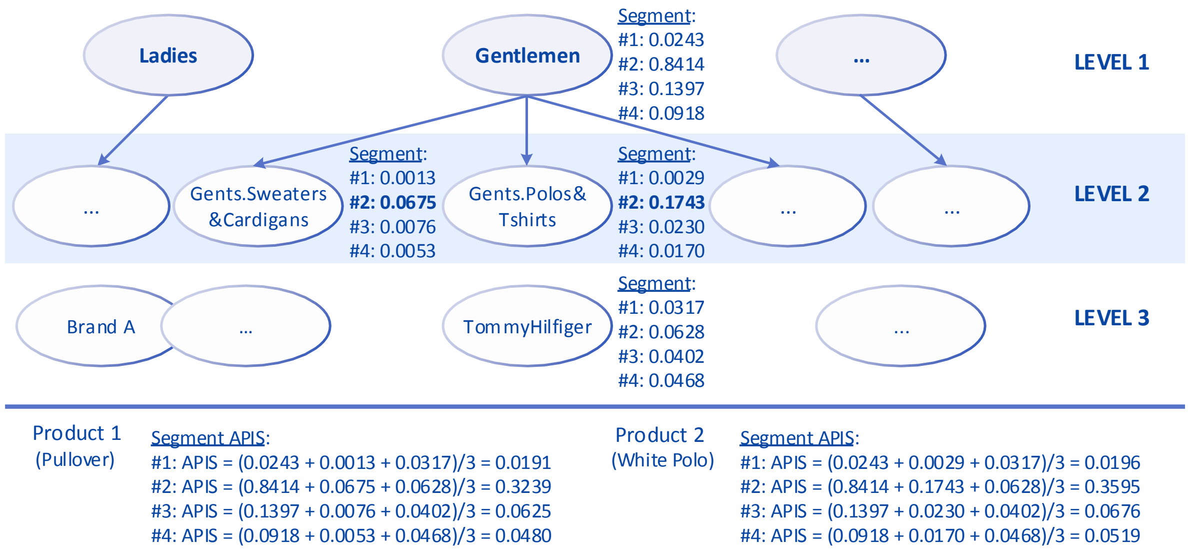

Figure 13: Segment specific average interest scores and segment-based product match rates (APIS).

Table 4: Min-max values and product-specific values for measures used to compute the final match rate of a product.

		segmentMatchRate	generalPopularity	newness	availability
Max	n/a	0.3677	1000	180	1
Min	n/a	0.0060	0	0	0
Control	Product 1	0	150	35	0.8
(default)	Product 2	0	75	7	0.6
Segment 2	Product 1	0.3239	200	35	0.8
	Product 2	0.3595	75	7	0.6

We now refer back to the two product lister pages depicted on Figures 11 and 12 for the search "tommy hilfiger". At the time of the experiment, 1,550 products satisfied this search condition. Figure 11 depicts the first nine products (out of 48) after a search for the brand "tommy hilfiger" for the default scenario (Control group). From Table 5 we see that *Product 1* (Pullover) has a final match rate of 0.5094, which has ranked it on the first position on the first lister page (Rank = 1), whereas *Product 2* (White Polo) with final match rate 0.4901 does not end on the first lister page.

In the case of segment-based re-ranking (Figure 12), we observe (Table 5) that *Product 2* (White Polo) scores much higher than *Product 1*, with final match rates 0.7293 and 0.6909, respectively, which places *Product 2* as the second product of the search result (Rank = 2, Figure

12). This segment-specific effect is caused by the fact that for Segment 2 the category *GentlemenPolos&Tshirts* on Level 2 is more popular (higher segment-specific match rate) than the category *GentlemenSweaters&Cardigans* (see Figure 13), and for this reason we can also observe 8 out 9 highest ranked products in Figure 12 to belong to the category *GentlemenPolos&Tshirts*. Only the first ranked product (Sneakers) does not belong to this Level 2 category but instead to *GentlemenSneakers*, with segment-specific Level 2 preference at value 0.1066, which is between the corresponding values for *Product 1* and *2* (Figure 13). The high rank of the "Sneakers" is due to very favourable values for the three other variables (generalPopularity, newness, and availability) in the final match rate computation.

The final result implies that segment-based search improves the efficiency of finding the final product(s) aligning the ultimate needs of the user, as indicated by commercial efficiency improvement discussed in the following section.

4.4. Commercial Evaluation

In order to investigate how the increasing average ranking of product clicks coincides with enhanced commercial efficiency, let us explore CTR, i.e., the number of product clicks observed per search, and $AtBR$, i.e., the number of add-to-basket events as a result of a search result

Table 5: Final product match rates for the default (control) and segment-based product listing scenarios. All values normalized.

	Control		Segment 2	
	Product 1	Product 2	Product 1	Product 2
α	0	0	0.5	0.5
segmentMatchRate	0	0	0.8789	0.9773
β	0.45	0.45	0.23	0.23
generalPopularity	0.15	0.075	0.15	0.075
γ	0.35	0.35	0.17	0.17
newness	0.8056	0.9611	0.8056	0.9611
$1-\alpha-\beta-\gamma$	0.2	0.2	0.1	0.1
availability	0.8	0.6	0.8	0.6
finalMatchRate	0.5094	0.4901	0.6909	0.7293

Table 6: The commercial results for the executed experiment.

		Searches	Product Clicks	Add-to-Baskets	CTR	AtBR
Setting 1	Treatment	19,435	2,665	248	0.137	0.093
	Control	5,986	867	83	0.145	0.096
Setting 2	Treatment	21,401	3,050	405	0.143	0.133
	Control	8,062	1,233	133	0.153	0.108
Setting 3	Treatment	17,804	2,440	284	0.137	0.116
	Control	6,276	910	97	0.145	0.107
Total	Treatment	58,640	8,155	937	0.139	0.115
	Control	20,324	3,010	313	0.148	0.104
Overall		78,964	11,165	1,250	0.141	0.112

product click. Table 6 lists the commercial results for the executed experiment over the three different settings (*Setting* 1: $\alpha = 0.2$, *Setting* 2: $\alpha = 0.5$, and *Setting* 3: $\alpha = 0.8$) for the treatment and the control groups ($\alpha = 0$), separately. The $AtBR$s are exclusively for products for which a product click was recorded on a search result lister page in the same session.

When comparing the CTR for the treatment and control groups, we see a decrease in the number of product clicks relative to the number of searches for all the three different settings (Table 6). This reduction in the number of clicks for the treatment group is explained by the fact that the treatment group lister page contains more relevant products together with their images and essential details for deciding on product suitability. For example, on the lister page depicted in Figure 12, we can observe a number of product attributes for each product – name, brand, description, price, and multiple images through mouse-over being directly provided on the search results page. In [10] it has been shown that inline summaries (in our case the product details given on the lister page together with images) add value to SERP, ease decision-making, and improve user performance. While going through the product lister page, users tend to click only on items that visually match their desired characteristics – thus users apply visual filtering and provided product attributes as cues for decisions [38] and only decide to open product detail pages (counted as clicks towards CTR) for items they find relevant [32] and expect to find value in, having a decreasing effect on CTR. This is also supported by the increase in average ranking of product clicks, as the incorporation of personalization has increased product findability, and thereby customers find more products of interest also lower on the results lister page and are not only satisfied with the top-most results, as common for Web search [21, 32, 49].

The comparison of the $AtBR$ for the treatment and control groups shows an increase in the add-to-basket events relative to the number of product clicks for *Setting* 2 and *Setting* 3. For *Setting* 1 the $AtBR$ is slightly lower than for the control group, which might be caused by the fact that segment-based match rate $\alpha = 0.2$ is not enough to boost the re-ranking of searched items for the product lister page – thus not having enough influence to affect the search re-ranking towards customer preferences. This is also in line with the previous section, where it was concluded that the relevance of the product suggestions was the least enhanced for *Setting* 1, compared to the performance for *Setting* 2 and *Setting* 3. Overall, introducing a segment-based match rate into the recommendation improves product findability, and increases the probability that a product is added to the shopping basket for purchase.

Next, we observe the differences in CTR and $AtBR$ between the three different treatment group settings and the control group observed over the same time span (shown in Figure 14). As seen, the difference in CTR relative to the control group decreases approximately equally for all three different settings, however, statistical significance, at a 95% confidence level, is not obtained. Furthermore, for *Settings* 2 and 3, the $AtBR$ increases relative to the control group, which implies that for $\alpha = 0.5$ and $\alpha = 0.8$ the probability of a product click to be converted to an add-to-basket event has increased. For $\alpha = 0.5$ this difference is even statistically significant since the entire 95% confidence interval of the difference is above zero. Together with the observation that fewer product clicks per search were observed, commercial efficiency improvement is concluded.

As we have shown, in a Web shop setting an increased value for the weight of the segment match rate α does not decrease the average ranking of product clicks (Figure 9); even more – the average ranking increases for higher values of α. Furthermore, from Figure 14 it is concluded that an increasing value for segment match rate α results in enhanced commercial efficiency of the search result lister page. Hence, an efficiency improvement of the search result product lister page for increasing values of α is found, since relative to the number of searches fewer product clicks are executed (lower CTR), while for these product clicks more add-to-basket events were observed, which is a desired phenomenon from the Web shop perspective. With this, the search

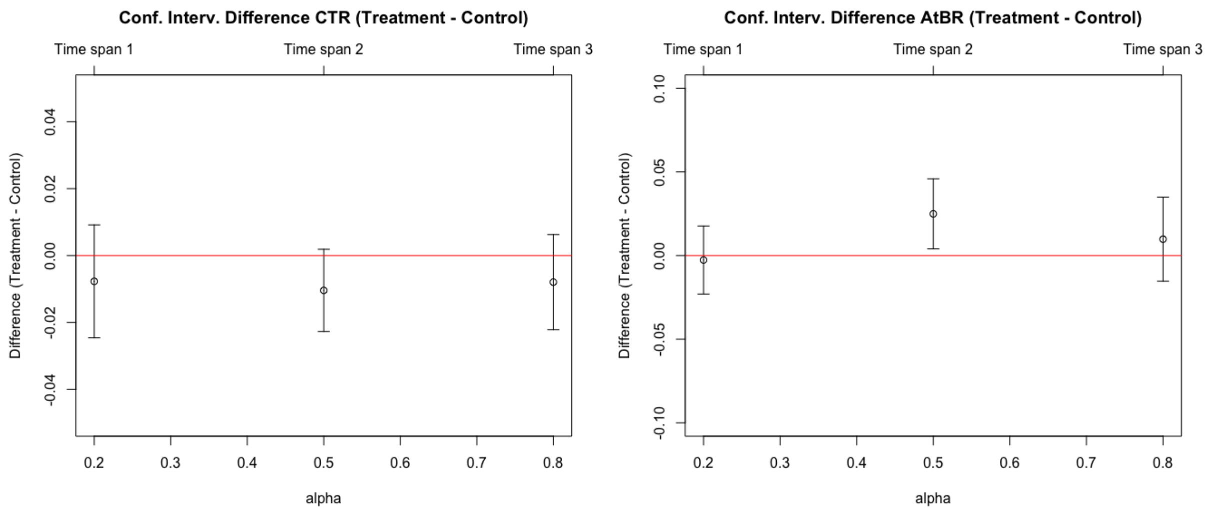

Figure 14: The 95% confidence intervals of the difference between the CTR of the treatment group and of the control group (left) and the difference in AtBR of the treatment group and of the control group (right) for the three different settings of α.

result product rankings were improved. On aggregate, the click-through rate and add-to-basket rate are the most optimal for $\alpha = 0.5$, meaning commercially an intermediate influence of the segment-based preferences yields the best results.

In summary, in the context of a Web shop product search engine a decrease in the average rank of product clicks is not necessarily an indication of an enhanced search result product lister page, i.e., does not deliver indication of lister page improvement. On the contrary, it was shown that incorporating segment-based product ranking on search result lister pages results in an increase in the average product click rank through improved product findability observed together with a decrease in the number of searches per session and per user for the treatment groups, as indicated in Table 3. Also, segment-based product ranking has shown the search result lister page to perform commercially more efficient, i.e., a lower click-through rate yet a higher add-to-basket rate, being in line with the findings of Jannach and Hegelich [29]. Moreover, incorporating segment-based relevance in the ranking of products and thus enhancing the relevance of products ranked high on the search result lister page, induces users to scroll down further, since more products on the search result page consequently match their personal preferences, lowers information overload [2], and improves trust into search results. Thus, the inclusion of segment-based relevance in the ranking brings an increased interest in the entire first lister page after a search as observed, and therefore an increase in the average rank of product clicks is recorded while improving the original non-semantic approach. This phenomenon is accompanied by an increased search efficiency.

5. Conclusion

In this research, we have proposed a methodology for re-ranking products on a Web shop search result lister page on a segment level supported by the assumption that products preferred by individuals with similar interests are an enrichment to a user's personal preferences. Our method consists of user profile construction using a modified existing product taxonomy of a Web shop as the profile schema, dimensionality reduction using PCA, k-means clustering, and re-ranking search results based on segment average preferences. User profiles were established based on product detail page views interpreted as implicit user feedback on their personal preferences. The segment compositions were used as the basis for the re-ranking procedure. A segment-based preference (i.e., match rate) for every product was determined and used supplementary to the existing influencing factors (popularity, newness, and availability) determining the final ranking of the products on a search result lister page. Dissimilar to current literature, search result rankings were altered on a segment level, instead of on an individual level, supported by the assumption that products preferred by individuals with similar interests are an enrichment to a user's personal preferences. In comparison to ontology-based search personalization, which is solely assessed for Web search engines, the proposed method is assessed for a Web shop product search engine, which in principle is different from Web document search. The method was implemented and evaluated with live experiments on re-ranking the search results based on segment-based preferences for De Bijenkorf's Web shop search engine with different (low, medium, and high) segment-based match rate influences. Although the experiments were carried out on De Bijenkorf's Web shop product search engine, the proposed solution is generic and one can easily apply the proposed solution to another Web shop using a different product category taxonomy.

Based on the empirical data, we conclude that counter-intuitively to what holds true for Web search, incorporating segment-based preferences in the ranking of products results in an increasing average rank of product clicks. A decrease in the number of product clicks from a search result lister page and the increased probability of product clicks to be converted to an add-to-basket event indicates an increased efficiency of the search result lister page from a commercial perspective. We also find that a search result lister page consisting of more products aligning with the user's personal preference increases the interest in the entire lister page, and thus makes the user scroll further down compared to the situation where less relevant products are shown. Surprisingly, the average rank of the product clicks – used as an efficiency indicator in Web search – turns out to be an unreliable indicator of the quality of the search result lister page for Web shop environments. This novel finding may well be connected to the differences between the users of Web search and the users (customers) of Web shop product search, the latter using the on-site search facilities simply to filter out suitable products according to their needs, and sometimes even with conflicting characteristics.

A promising idea for further research is to investigate whether segment-based search can act superior to personalized search. The rationale is in the assumption that personalization does not incorporate information transfer between like-minded individuals, and on a large scale, personalization is computationally-intensive for Web shop search engines. Also, simultaneous testing with different measures for the final match rate would be more robust to the volatility of external and internal factors and would allow to compare different settings directly providing enriching insights into product ranking. As user preferences change over time, it would be interesting to study the user profile adjustment frequency (e.g., weekly, every few weeks, or months) and its effect on search result ranking. In addition, we would like to investigate other clustering approaches next to k-means like k-medoids clustering and k-harmonic means clustering. Lastly, although this study was about re-ranking on-site search engine results in Web shop environment through semantic segmentation of user profiles, it would be interesting to also compare the proposed method to different recommendation approaches in the real-life setting of some Web shop.

References

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional space. In *8th International Conference on Database Theory (ICDT 2001)*, volume 1973 of *Lecture Notes in Computer Science*, pages 420–434. Springer, 2001. doi: https://doi.org/10.1007/3-540-44503-X_27.

[2] M. Aljukhadar, S. Senecal, and C.-E. Daoust. Using recommendation agents to cope with information overload. *International Journal of Electronic Commerce*, 17:41–70, 12 2012. doi: https://doi.org/10.2307/41739511.

[3] R. A. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval.* Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1999. ISBN 020139829X.

[4] J. Bhogal, A. Macfarlane, and P. Smith. A review of ontology based query expansion. *Information Processing & Management*, 43(4):866–886, 2007. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2006.09.003.

[5] A. Bouhana, A. Zidi, A. Fekih, H. Chabchoub, and M. Abed. An ontology-based CBR approach for personalized itinerary search systems for sustainable urban freight transport. *Expert Syst. Appl.*, 42(7):3724–3741, May 2015. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2014.12.012.

[6] S. Calegari and G. Pasi. Personal ontologies: Generation of user profiles based on the yago ontology. *Information Processing & Management*, 49(3):640–658, 2013. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2012.07.010.

[7] T. Caliński and J. Harabasz. A dendrite method for cluster analysis. *Communications in Statistics-Simulation and Computation*, 3(1):1–27, 1974. ISSN 0090-3272. doi: https://doi.org/10.1080/03610927408827101.

[8] A. W. Ding, S. Li, and P. Chatterjee. Learning user real-time intent for optimal dynamic web page transformation. *Info. Sys. Research*, 26(2):339–359, Jun 2015. ISSN 1526-5536. doi: https://doi.org/10.1287/isre.2015.0568.

[9] H. Drachsler, H. G. K. Hummel, and R. Koper. Identifying the goal, user model and conditions of recommender systems for formal and informal learning. *J. Digit. Inf.*, 10, 2009. ISSN 1368-7506.

[10] S. Dumais, E. Cutrell, and H. Chen. Optimizing search by showing results in context. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, page 277–284, New York, NY, USA, 2001. Association for Computing Machinery. doi: https://doi.org/10.1145/365024.365116.

[11] Fang Liu, C. Yu, and Weiyi Meng. Personalized web search for improving retrieval effectiveness. *IEEE Transactions on Knowledge and Data Engineering*, 16(1):28–40, 2004. ISSN 2326-3865. doi: https://doi.org/10.1109/TKDE.2004.1264820.

[12] E. Forgy. Cluster analysis of multivariate data: Efficiency versus interpretability of classification. *Biometrics*, 21:768–769, 1965.

[13] V. Formoso, D. Fernandez, F. Cacheda, and V. Carneiro. Using profile expansion techniques to alleviate the new user problem. *Information Processing & Management*, 49(3):659–672, 2013. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2012.07.005.

[14] R. Gao and C. Shah. Toward creating a fairer ranking in search engine results. *Information Processing & Management*, 57(1):102138, 2020. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2019.102138.

[15] F. Garcia-Sanchez, R. Colomo-Palacios, and R. Valencia-Garcia. A social-semantic recommender system for advertisements. *Information Processing & Management*, 57(2):102153, 2020. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2019.102153.

[16] S. Gauch, J. Chaffee, and A. Pretschner. Ontology-based personalized search and browsing. *Web Intelli. and Agent Sys.*, 1(3-4):219–234, 2003. ISSN 1570-1263. doi: https://doi.org/10.5555/1016416.1016421.

[17] S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli. *User Profiles for Personalized Information Access*, pages 54–89. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. ISBN 9783540720799. doi: https://doi.org/10.1007/978-3-540-72079-9_2.

[18] T. Geurts, S. Giannikis, and F. Frasincar. Active learning strategies for solving the cold user problem in model-based recommender systems. *Web Intell.*, 18(4):269–283, 2020. doi: https://doi.org/10.3233/WEB-210448.

[19] S. Goel, R. Kumar, M. Kumar, and V. Chopra. An efficient page ranking approach based on vector norms

using snorm(p) algorithm. *Information Processing & Management*, 56(3):1053–1066, 2019. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2019.02.004.

[20] J. H. Goldberg, M. J. Stimson, M. Lewenstein, N. Scott, and A. M. Wichansky. Eye tracking in web search tasks: Design implications. In *Proceedings of the 2002 Symposium on Eye Tracking Research & Applications*, ETRA '02, page 51–58. Association for Computing Machinery, 2002. doi: https://doi.org/10.1145/507072.507082.

[21] Z. Guan and E. Cutrell. An eye tracking study of the effect of target rank on web search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, page 417–420, New York, NY, USA, 2007. Association for Computing Machinery. doi: https://doi.org/10.1145/1240624.1240691.

[22] J. Hartigan and M. Wong. Algorithm AS 136: A K-means clustering algorithm. *Applied Statistics*, pages 100–108, 1979. ISSN 0035-9254. doi: https://doi.org/10.2307/2346830.

[23] J. Hautala, C. Kiili, Y. Kammerer, O. Loberg, S. Hokkanen, and P. H. T. Leppänen. Sixth graders' evaluation strategies when reading internet search results: an eye-tracking study. *Behaviour & Information Technology*, 37 (8):761–773, 2018. doi: 10.1080/0144929X.2018.1477992.

[24] S. Y. Ho, D. Bodoff, and K. Y. Tam. Timing of adaptive web personalization and its effects on online consumer behavior. *Information Systems Research*, 22(3):660–679, 2011. ISSN 1526-5536. doi: https://doi.org/10.1287/isre.1090.0262.

[25] L.-p. Hung. A personalized recommendation system based on product taxonomy for one-to-one marketing online. *Expert Systems with Applications*, 29(2):383–392, 2005. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2005.04.016.

[26] N. Ibrahim, A. H. Chaibi, and H. B. Ghézala. Scientometric re-ranking approach to improve search results. *Procedia Computer Science*, 112:447–456, 2017. ISSN 1877-0509. doi: https://doi.org/10.1016/j.procs.2017.08.020.

[27] P. Indyk and R. Motwani. Approximate nearest neighbors: Towards removing the curse of dimensionality. In *Proceedings of the Thirtieth Annual ACM Symposium on Theory of Computing*, STOC '98, page 604–613, New York, NY, USA, 1998. Association for Computing Machinery. doi: https://doi.org/10.1145/276698.276876.

[28] V. Janev and S. Vranes. Applicability assessment of semantic web technologies. *Information Processing & Management*, 47(4):507–517, 2011. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2010.11.002.

[29] D. Jannach and K. Hegelich. A case study on the effectiveness of recommendations in the mobile internet. In *Proceedings of the Third ACM Conference on Recommender Systems*, RecSys 2009, pages 205–208, New York, NY, USA, 2009. Association for Computing Machinery. doi: https://doi.org/10.1145/1639714.1639749.

[30] S. Jiang and C. E. Beaudoin. Health literacy and the Internet: An exploratory study on the 2013 hints survey. *Computers in Human Behavior*, 58:240–248, 2016. ISSN 0747-5632. doi: https://doi.org/10.1016/j.chb.2016.01.007.

[31] T. Joachims. Optimizing search engines using clickthrough data. In *Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD 2002, pages 133–142, New York, NY, USA, 2002. ACM. doi: https://doi.org/10.1145/775047.775067.

[32] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and*

Development in Information Retrieval, SIGIR '05, page 154–161, New York, NY, USA, 2005. Association for Computing Machinery. doi: https://doi.org/10.1145/1076034.1076063.

[33] K. Juszczyszyn, P. Kazienko, and K. Musiał. Personalized ontology-based recommender systems for multimedia objects. In *Agent and Multi-agent Technology for Internet and Enterprise Systems*, pages 275–292. Springer, 2010. ISBN 9783642135262. doi: https://doi.org/10.1007/978-3-642-13526-2_13.

[34] P. Kouki, I. Fountalis, N. Vasiloglou, N. Yan, U. Ahsan, K. A. Jadda, and H. Qu. Product collection recommendation in online retail. In *Proceedings of the 13th ACM Conference on Recommender Systems*, RecSys 2019, pages 486—490, New York, NY, USA, 2019. ACM. doi: https://doi.org/10.1145/3298689.3347003.

[35] N. Kumar and K. R. Lang. Do search terms matter for online consumers? The interplay between search engine query specification and topical organization. *Decision Support Systems*, 44(1):159–174, 2007. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2007.03.010.

[36] S. Levien. Forrester: US mobile and tablet commerce to top $293b by 2018, total ecommerce to hit $414. Technical report, Forrester Research, 2014. URL `http://goo.gl/O1R4dS`.

[37] B. Li, A. Ghose, and P. G. Ipeirotis. Towards a theory model for product search. In *Proceedings of the 20th International Conference on World Wide Web, (WWW 2011)*, pages 327–336, New York, NY, USA, 2011. ACM. doi: https://doi.org/10.1145/1963405.1963453.

[38] C.-Y. Li. Why do online consumers experience information overload? an extension of communication theory. *Journal of Information Science*, 43(6):835–851, 2017. ISSN 0165-5515. doi: https://doi.org/10.1177/0165551516670096.

[39] S. Lloyd. Least squares quantization in PCM. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982. ISSN 1557-9654. doi: https://doi.org/10.1109/TIT.1982.1056489.

[40] L. Lorigo, M. Haridasan, H. Brynjarsdóttir, L. Xia, T. Joachims, G. Gay, L. Granka, F. Pellacini, and B. Pan. Eye tracking and online search: Lessons learned and challenges ahead. *Journal of the American Society for Information Science and Technology*, 59(7):1041–1052, 2008. ISSN 2330-1635. doi: https://doi.org/10.1002/asi.20794.

[41] R. Lucian. Digital overload: The effects of the large amounts of information when purchasing online. *Journal of Internet Banking and Commerce*, 19(2):1–18, 2014. ISSN 1204-5357.

[42] J. Macqueen. Some methods for classification and analysis of multivariate observations. In *5th Berkeley Symposium on Mathematical Statistics and Probability (BSMSP 1967)*, pages 281–297. University of California Press, 1967.

[43] A. Moreno, A. Valls, D. Isern, L. Marin, and J. Borrís. Sigtur/e-destination: Ontology-based personalized recommendation of tourism and leisure activities. *Engineering Applications of Artificial Intelligence*, 26(1):633–651, 2013. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2012.02.014.

[44] M. Nakatsuji, M. Yoshida, and T. Ishida. Detecting innovative topics based on user-interest ontology. *Journal of Web Semantics*, 7(2):107–120, 2009. ISSN 1570-8268. doi: https://doi.org/10.1016/j.websem.2009.01.001.

[45] R. Navigli. Word sense disambiguation: A survey. *ACM Computing Surveys*, 41(2):10:1–10:69, 2009. ISSN 0360-0300. doi: https://doi.org/10.1145/1459352.1459355.

[46] L. J. Nederstigt, S. S. Aanen, D. Vandic, and F. Frasincar. FLOPPIES: A framework for large-scale ontology

population of product information from tabular data in e-commerce stores. *Decision Support Systems*, 59: 296–311, 2014. ISSN 0167-9236. doi: https://doi.org/10.1016/j.dss.2014.01.001.

[47] W. Ng, L. Deng, and D. L. Lee. Mining user preference using spy voting for search engine personalization. *ACM Trans. Internet Technol.*, 7(4), 2007. ISSN 1533-5399. doi: https://doi.org/10.1145/1278366.1278368.

[48] M. G. H. Omran, A. P. Engelbrecht, and A. Salman. An overview of clustering methods. *Intelligent Data Analysis*, 11(6):583–605, 2007. ISSN 1088-467X. doi: https://doi.org/10.3233/IDA-2007-11602.

[49] B. Pan, H. Hembrooke, T. Joachims, L. Lorigo, G. Gay, and L. Granka. In google we trust: Users' decisions on rank, position, and relevance. *Journal of Computer-Mediated Communication*, 12(3):801–823, 2007. doi: https://doi.org/10.1111/j.1083-6101.2007.00351.x.

[50] G. Punj and D. W. Stewart. Cluster analysis in marketing research: Review and suggestions for application. *Journal of Marketing Research*, 20(2):134–148, 1983. ISSN 1547-7193. doi: https://doi.org/10.2307/3151680.

[51] F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor. *Recommender Systems Handbook*. Springer-Verlag, 2011. ISBN 9780387858197.

[52] P. G. Roetzel. Information overload in the information age: a review of the literature from business administration, business psychology, and related disciplines with a bibliometric approach and framework development. *Business Research*, 12(2):479–522, 2019. ISSN 2198-2627. doi: https://doi.org/10.1007/s40685-018-0069-z.

[53] M. Salehi and I. N. Kamalabadi. Hybrid recommendation approach for learning material based on sequential pattern of the accessed material and the learner's preference tree. *Knowledge-Based Systems*, 48:57–69, 2013. ISSN 0950-7051. doi: https://doi.org/10.1016/j.knosys.2013.04.012.

[54] S. Salvador and P. Chan. Determining the number of clusters/segments in hierarchical clustering/segmentation algorithms. In *16th IEEE International Conference on Tools with Artificial Intelligence (ICTAI 2004)*, pages 576–584. IEEE Computer Society, 2004. doi: https://doi.org/10.1109/ICTAI.2004.50.

[55] P. Sanchez and A. Bellogin. Building user profiles based on sequences for content and collaborative filtering. *Information Processing & Management*, 56(1):192–211, 2019. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2018.10.003.

[56] M. Sato, H. Izumo, and T. Sonoda. Discount sensitive recommender system for retail business. In *Proceedings of the 3rd Workshop on Emotions and Personality in Personalized Systems 2015*, EMPIRE 2015, pages 33–40, New York, NY, USA, 2015. ACM. doi: https://doi.org/10.1145/2809643.2809646.

[57] A. Sieg, B. Mobasher, and R. D. Burke. Learning ontology-based user profiles: A semantic approach to personalized web search. *IEEE Intelligent Informatics Bulletin*, 8(1):7–18, 2007. ISSN 1727-5997.

[58] M. Speretta and S. Gauch. Personalized search based on user search histories. In *2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI 2005)*, pages 622–628. IEEE Computer Society, 2005. doi: https://doi.org/10.1109/WI.2005.114.

[59] P.-N. Tan, M. Steinbach, A. Karpatne, and V. Kumar. *Introduction to Data Mining*. Pearson, 2nd edition, 2018. ISBN 0133128903.

[60] M. Verleysen and F. Damien. The curse of dimensionality in data mining and time series prediction. In *8th International Work-Conference on Artificial Neural Networks (IWANN 2005)*, volume 3512 of *Lecture Notes in Computer Science*, pages 758–770. Springer, 2005. doi: https://doi.org/10.1007/11494669_93.

[61] S.-S. Weng and M.-J. Liu. Feature-based recommendations for one-to-one marketing. *Expert Systems with Applications*, 26(4):493–508, 2004. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2003.10.008.

[62] L. Wu, D. Hu, L. Hong, and H. Liu. Turning clicks into purchases: Revenue optimization for product search in e-commerce. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, SIGIR 2018, pages 365–374, New York, NY, USA, 2018. ACM. doi: https://doi.org/10.1145/3209978.3209993.

[63] W. Zhao, R. Wu, and H. Liu. Paper recommendation based on the knowledge gap between a researcher's background knowledge and research target. *Information Processing & Management*, 52(5):976–988, 2016. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2016.04.004.

[64] C.-N. Ziegler, G. Lausen, and L. Schmidt-Thieme. Taxonomy-driven computation of product recommendations. In *Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management*, CIKM '04, page 406–415, New York, NY, USA, 2004. Association for Computing Machinery. doi: https://doi.org/10.1145/1031171.1031252.

Appendix A

This appendix provides additional insights and technical background information on user profiles and the interest scores, related to the discussion in Section 3.1.

In Section 3.1 we explained that a user profile is expressed as a vector of weighted interest scores corresponding to all concepts in the taxonomy (Equation 2). To exemplify the construction of user profiles, let us consider a new user that has shown interest in ladies trousers of *Brand A* and a ladies jacket of *Brand B*. For a new user, all the interest scores are initially 0. An illustration of how this is recorded initially in terms of interest scores (IS) and transformed into a weighted concept hierarchy is visualized in Figure 15.

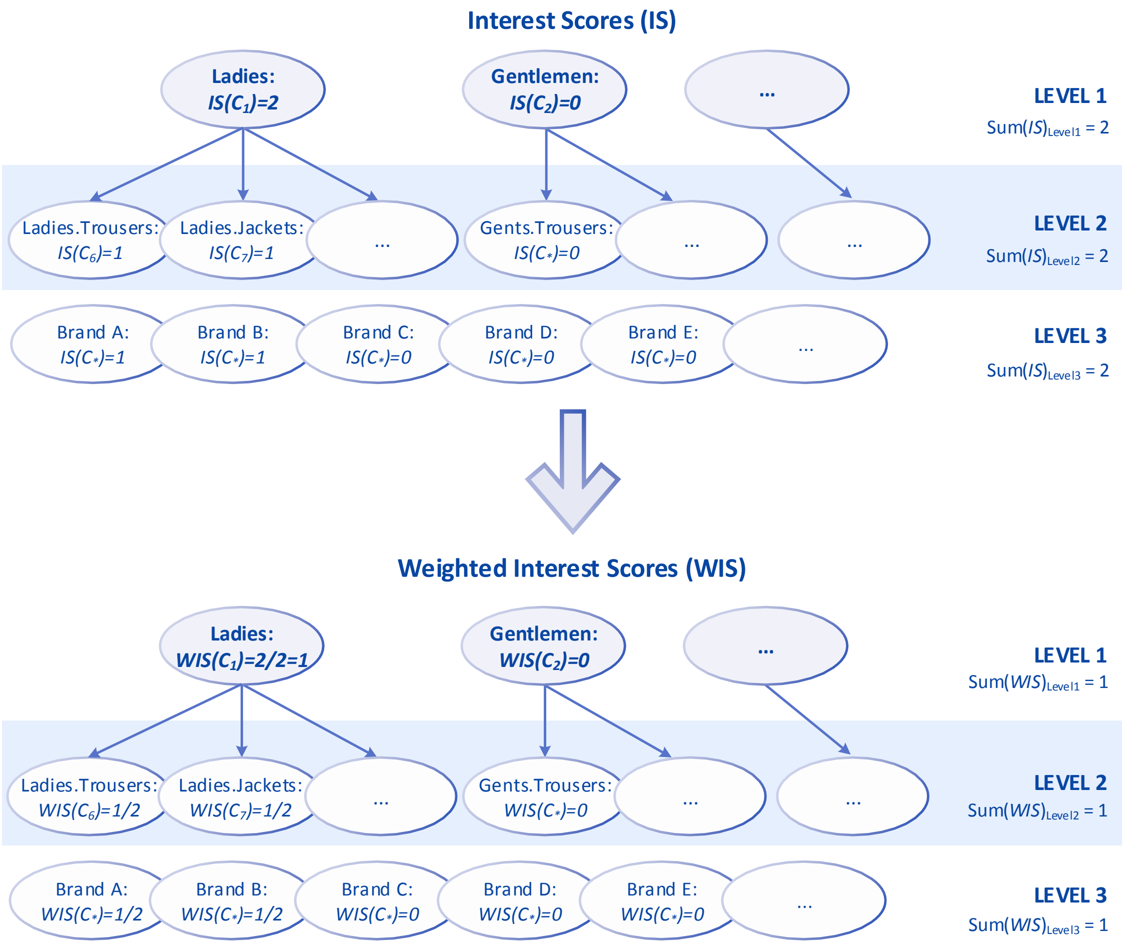

Figure 15: Example of user profile construction, illustrating update to interest scores of a new user who viewed a pair of ladies trousers of *Brand A* and a ladies jacket of *Brand B*.

First, in every level of the taxonomy, the page views are categorized such that for every page view the interest score of one concept in every level is incremented by one. When all interest scores have been categorized, the weighted interest scores are determined by dividing the interest scores of every concept in a level by the total sum of interest scores of that level. This sets the weighted interest score sum to 1 for every level in the taxonomy. This results in the determination of the relative importance of every concept to all other concepts in the concerning level, represented by the weighted interest scores (WIS).

Appendix B

This appendix delivers supplementary data on the relative importance of concepts on *Level 2* and *Level 3* of the modified product taxonomy in addition to what was presented for *Level 1* in Table 1 from Section 3.2. In Tables 7 and 8 one can observe the average weighted interest scores for the six most popular concepts for all four segments in *Level 2* and *Level 3* of the hierarchy, respectively.

Table 7: The six most popular concepts in level 2 of the modified product taxonomy for all segments.

Concepts	L.Dresses	L.Tops	NA.NA	L.Sneakers	L.Swimwear	L.Sweaters
Segment 1	0.183	0.112	0.064	0.049	0.044	0.038
Concepts	G.Polo's T-shirts	G.Sneakers	G.Jackets	G.Sweaters	G.Jeans	G.Shirts
Segment 2	0.167	0.107	0.095	0.068	0.059	0.050
Concepts	C.Tops	NA.NA	C.Sweaters	C.Jackets	H.Tableware	C.Trousers
Segment 3	0.096	0.075	0.034	0.033	0.032	0.031
Concepts	NA.NA	L.Dresses	L.Tops	L.Sneakers	L.Swimwear	L.Lingerie
Segment 4	0.105	0.078	0.071	0.038	0.029	0.026

L - Ladies; G - Gents; C - Children; H - Houseware;

Table 8: The six most popular concepts in level 3 of the modified product taxonomy for all segments.

Concepts	TED BAKER	MICHAEL KORS	TOMMY HILFIGER	PHASE EIGHT	WAREHOUSE	KAREN MILLEN
Segment 1	0.043	0.032	0.032	0.026	0.024	0.022
Concepts	STONE ISLAND	TOMMY HILFIGER	POLO RALPH LAUREN	HUGO BOSS	SUNGLASS HUT	CALVIN KLEIN
Segment 2	0.088	0.063	0.040	0.034	0.034	0.032
Concepts	Miscellaneous Brands	TOMMY HILFIGER	STRATING CC	STONE ISLAND	BENETTON	KENZO
Segment 3	0.064	0.047	0.030	0.023	0.022	0.020
Concepts	TOMMY HILFIGER	Miscellaneous Brands	SUNGLASS HUT	TED BAKER	MICHAEL KORS	CALVIN KLEIN
Segment 4	0.040	0.030	0.022	0.022	0.020	0.019