Journal of Web Engineering, Vol. 8, No. 2 (2009) 113-153

ONTOLOGY-DRIVEN PERSONALIZED QUERY REFINEMENT

SOFIA STAMOU LEFTERIS KOZANIDIS PARASKEVI TZEKOU NIKOS ZOTOS

Computer Engineering and Informatics Department, Patras University, Greece
{stamou, kozanid, tzekou, zotosn}@ceid.upatras.gr

Received July 2, 2007
Revised October 29, 2008

The most popular way for finding information on the Web is go to a search engine, submit a query that describes an information need and receive a list of results that relate to the information sought. As more and more topics are being discussed over the Web and our vocabulary remains relatively stable, it is increasingly difficult for Web users to select queries that express their varying information needs in a distinguishable by the engine manner. *Query refinement* is the process of providing information seekers with alternative wordings for expressing their search intentions. Although refined queries may contribute to the improvement of retrieval results, nevertheless their realization is intrinsically limited in that they consider nothing about the preferences of the user issuing that query. One way to go about selecting suitable query alternatives is to account for the user interests in the query refinement process. This task involves two great challenges. First we need to be able to effectively identify the user preferences and build a profile for every user. Second, once such a profile is available, we need to identify among a set of candidate query alternatives those that match the user interests. In this article, we present our work towards a personalized query refinement technique and we discuss how we address both of these challenges. Since Web users are reluctant to provide explicit information on their personal preferences, for the first challenge we attempt to determine them based on the analysis of the users' click history. In particular, we leverage a topical ontology for estimating the user's topic preferences based on her past searches. For the second challenge, we have developed a query refinement mechanism that uses the learnt user preferences in order to disambiguate the user's current query and thereafter identify alternative query wordings that match both the initial query semantics and the user preferences. Our experiments show that user preferences can be learnt accurately through the use of the topical ontology and refined queries based on the user preferences yield significant improvements in the search quality over existing query improvement techniques.

Key words: Personalized search, user preferences, topic-specific rankings, query refinement, topical ontology
Communicated by: J. Hoben & R. Baeza-Yates

1 Introduction

Thousands of users issue keyword queries to search engines daily for locating information about a multitude of topics. Since Web users may have diverse backgrounds and goals in mind when querying the Web, it is of paramount importance that the keywords they select for describing their information needs are reflective of their underlying search intentions. However, a number of studies have shown that a significant number of queries to search engines are under-specified and contain only a few words [18]. Short queries, being marginally informative of the users' search intentions, result oftentimes to the retrieval of search results that might not satisfy the users' information needs. To make things worse, queries might be polysemous; using identical terms to represent distinct information needs, making therefore the retrieval of relevant data arduous.

To address the user difficulties in finding relevant information about a topic of interest there has been a great body of work, which mainly falls into two distinct yet complementary directions: *query refinement* and *search personalization*. Query refinement is the process of proving users with alternative query formulations in the hope of retrieving relevant information. On the other hand, search personalization is the process of incorporating information about the user in the query processing phase in the hope of retrieving results that satisfy the user intentions. Although query refinement may help users specify detailed queries, nevertheless it considers nothing about the user intentions. As such, it delivers the same alternatives for a query regardless of the varying search intentions that are hidden behind that query. Conversely, search personalization aims at learning the latent user preferences that are hidden behind search queries and based on this knowledge it retrieves results that pertain to those preferences. However, personalization has not been adequately addressed in the context of explicit query improvement.

In this article, we attempt to fill this void and we study the problem of personalized query refinement. To exemplify the pursuit of our study consider the following scenario. User A, an IT specialist, submits the query *printer* to a search engine and wants to look for information about a printer device. Additionally, user B, a novelist, submits the same query (*printer*) to the same engine, in order to look for a printing house to typeset his novel. If the search engine that our users employ incorporates personalization techniques it will retrieve for user A pages that talk about printer devices, while for user B it will retrieve pages that contain information about companies or people whose occupation is printing. Suppose now that none of our users has found the exact information she was looking for in the first few (personalized) results returned for her query. The most likely reaction of our users would be either to submit a new query (i.e. start a new search session) or quit searching in that engine. In both cases though, the user would encounter a failed search experience. However, this failure is not attributed to the engine's difficulty in answering the user queries, but rather on the users' lacking ability to specify detailed information needs.

Suppose now that the search engine was equipped with a personalized query refinement mechanism. Upon query issuing it would not only retrieve personalized results but it would also suggest the users alternative query formulations that relate to their search intentions. Therefore, our users could employ any of the system suggested terms to improve their current searches instead of starting new ones. In such case, the engine would suggest user A the alternative queries: *color printer*, *laser printer*, *inkjet*, etc. and user B the alternative queries: *pressman*, *publisher*, *printmaker*. By providing users with personalized refined queries, we believe that a search engine will have increased usability and user retention, while at the same time the users will experience better Web searches.

Personalized query refinement is the process of dynamically improving a query with related terms that are informative of and tailored to the information needs of specific users. This task involves two great challenges. First we need to be able to effectively identify the user preferences and build a profile for every user. Second, once such a profile is available, we need to identify among a set of candidate query alternatives those that match the user interests. In our work, we introduce the use of a topical ontology for enabling personalized query refinement. In particular, we leverage the ontology for learning a user's search profile, based on the topical analysis of her previous searchers. Based on this profile, we disambiguate the user's current query by relying on the semantic correlation between the query keywords and the user topic preferences. Having disambiguated the user's query, we rely on the ontology for identifying alternative terms that match the query semantics. Finally, we select among the query alternatives those that maximize their relatedness to the user learnt preferences in order to refine that user's search request. Our experiments show that user preferences can be learnt accurately through the use of the topical ontology and refined queries based on the user preferences yield significant improvements in the search quality over existing query improvement techniques.

Although, there have been previous studies that address the problem of personalization in the context of both database [28] and Web searching [31], most of these efforts concentrate on mapping queries to categories that are likely to be related to the user interests. These categories serve as a context to disambiguate the words in the user's query in order to enable search personalization. Our study is different from existing works in that we go beyond search personalization and we study whether and how we can personalize the query refinement process. The underlying objective of our study is not merely to associate queries to user-preferred categories but also to recommend users alternative query wordings that relate to both their initial queries and their preferred topics. In brief, the contributions of this article are as follows.

- In Section 3, we present a method that automatically identifies the hidden search preferences of a user, based on the topical analysis of her observable past clickthrough data. Unlike existing techniques that operate upon pre-classified pages for learning the user interests, our method uses a topical ontology for automatically annotating the user's click history with topical categories. Therefore, our method can be successfully applied to pages with yet-unknown topics (such as dynamic pages). Based on the topical analysis of the user's click history, we estimate the user's degree of preference in different topics. Through the use of the ontology in the user profiling process, we ensure that our model is generic enough to accommodate volatile user preferences, i.e. the case that a user's interests change over time.
- In Section 4, we use the learnt user preferences to disambiguate the semantics of a current query for which there is no click data available. We then show how to use the ontology to identify alternative query terms that match both the query semantics and the user interests. In contrast to existing query refinement approaches that reach up to the extend of resolving query sense ambiguities, in our work we take a step further and we investigate how we can make a query refinement technique personalized.
- In Section 5, we address the challenge of visualizing the personalized queries, in order to assist web users realize the suitability of the system selected terms in describing their search intentions. For queries' visualization we adopt a graphical representation scheme, which structures the improved queries into personalized topical graphs and enables the user interaction with them. By doing so, we assist the user make informed decisions about what query to submit next in case her information need is not fully met by her self-selected keyword queries.
- In Section 6, we present our experimental evaluation, which compares the retrieval performance of our personalized query refinement technique with other well-known query improvement methods. We also present a user survey we carried out in order to evaluate the learning accuracy of our user interests' identification approach.

Finally, Section 7 discusses related work, while Section 8 provides further discussion and concludes the article.

2 Background on Ontology-Based Web Page Annotations

This section introduces the notation and necessary background for this article. We first present the topical ontology that we explore in our study for enabling personalized query refinement and briefly summarize how the ontology can serve towards the annotation of the Web pages' contents with a suitable topical category. Then, we review how to estimate the pages' importance to the assigned ontology topics via a topic-importance ranking function.

2.1 The Topical Ontology

For our purpose of using an ontology to identify the general topics that might be of interest to the Web users, we choose to develop an ontology that would describe humans' perception of the most popular topics that are communicated in the Web data. Thus, we define our ontology as a hierarchy of topics that are currently used for categorizing Web pages. To ensure that our ontology would define concepts that are representative of the Web's topical content, we borrowed the ontology's top level concepts from the topic categories of the Dmoz Directory[a]. Moreover, to guarantee that our ontology would be of good quality, we obtained our ontology's conceptual hierarchies from existing ontological resources that have proved to be richly encoded and useful. The knowledge bases that we explored for building the ontology are the Suggested Upper Merged Ontology (SUMO)[b], the MultiWordNet Domains (MWND) and WordNet 2.0[c]. WordNet is a lexical network of more than 115K synonym sets (synsets) that are linked to other synsets on the basis of their semantic properties and/or features. MWND is an augmented version of WordNet; a resource hat assigns every WordNet[d] synset a domain label among the total set of 165 hierarchically structured domains it consists of. SUMO is a generic ontology of more than 1,000 domain concepts that have been mapped to every WordNet synset that is related to them. The rationale for exploring the above resources to build our ontology is their acknowledged rich and qualitative content and the fact that they are mapped to a common lexical network, i.e. WordNet.

For a thorough description of the methodology we adopted for building the ontology, we refer the interested reader to the work of [38]. In brief, the construction of our ontology involved anchoring to the Dmoz top level categories, sub-topics (taken from either SUMO or MWND) that represent the ontology's middle level concepts. Middle level concepts were determined after merging MWND and SUMO into a single combined resource. Merging SUMO hierarchies and MWND domains into a common ontology was carried out manually following an iterative process, during which WordNet hierarchies were regularly consulted. The first straightforward step we took was to detect common domain labels across the two resources. Domains of identical or quasi-identical names were traced and all their corresponding WordNet hierarchies were retrieved and cross-checked. Hierarchies with a suf-

[a] http://dmoz.org
[b] http://ontology.teknowledge.com
[c] http://www.cogsci.princeton.edu/~wn/
[d].MWND labels were originally assigned to WordNet 1.6 synsets, but we augmented them to WordNet 2.0 using the mappings (available at http://www.cogsci.princeton.edu/~wn/links.shtml) between the different WordNet versions

ficient number of overlapping elements were merged together into their common parent domain, using their WordNet relations. This parent concept was then searched in the Dmoz top level topics. If there was a matching found between a top level topic and the parent concept of a combined hierarchy, the latter was integrated with this top level domain.

Conversely, if no matching was found, the direct hypernyms of the parent concept were retrieved from WordNet and searched within the top level topics. If there was a matching between the hypernyms of a merged hierarchy's parent concept and a top level concept, the hierarchy was integrated with the top level domain through a specialization (IS-A) relation. This way the joint hierarchy's parent concept becomes a sub-topic in one of the top level categories and denotes a middle level concept. In our model we treat the middle level concepts as subtopics. At the end of this process, we had merged a significant number of hierarchies into the top level domains, but still there were hierarchies left disjoint. We manually partitioned these disjoint hierarchies into two classes. The first class contained the hierarchies whose domain concepts are semantically related in WordNet and the second class contained everything else. The hierarchies of the second class were disregarded because there was not sufficient evidence in WordNet to support our judgments for their potential merging. Hierarchies of the first class were merged together as it is communicated below.

Every domain concept, (either form SUMO or MWND) was searched sequentially in WordNet and all its hypernyms two levels up in the hierarchy were retrieved. Retrieved hypernyms were searched among the ontology's top and middle level domain concepts, following the same process as before. If a (co)-hypernym of a disjoint hierarchy matched a top or middle level concept; the former was incorporated in the ontology as a middle level concept and had its corresponding WordNet hierarchies appended as lower level concepts. Upon failure to find a matching upper or middle level concept, merging was terminated and the hierarchies remaining disjoint were omitted from the ontology. We limited the number of hypernyms considered to two, instead of continuing going up in WordNet hierarchies, to weed out too abstract concepts from the ontology. The intuition for restricting the upwards traversal of WordNet hierarchies to two level hypernyms is that the higher a concept is in a hierarchy the greater the likelihood that it is a coarse grained concept that may lead to obscure distinctions about the pages' topics. By imposing this limitation, we anticipate that concepts in coarser grain are pruned form the ontology's middle level concepts.

At the end of this process, we came down to a total set of 489 middle level concepts, which were organized to the 15 Dmoz top level topics, using their respective WordNet relations. The resulting upper ontology (i.e. top and middle level concepts) is a directed acyclic graph with maximum depth 6 and maximum branching factor, 28 (i.e. number of children concepts from a node). Figure 1 shows a portion of our ontology for the Dmoz topic Society.

Finally, we anchored to each middle level concept its corresponding WordNet hierarchies, the lexical elements of which are the ontology's lower level concepts. Anchoring was conducted semi-automatically, firstly by integrating all WordNet hierarchies that encounter a specialization link to any of the ontology's middle level concepts. Then, we manually verified or corrected each hierarchy anchored underneath the middle level concepts. This way, we confirmed that the ontology comprises qualitative rather than exhaustive hierarchies.

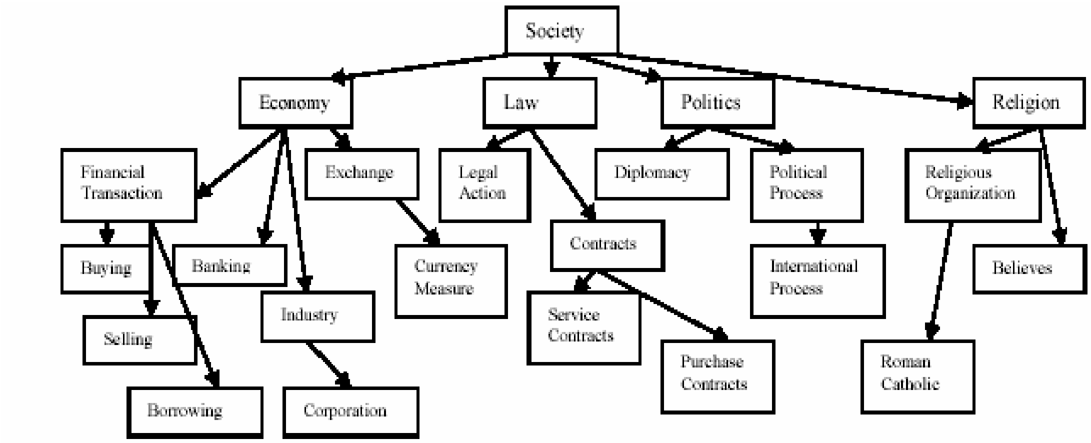

Figure 1. A portion of the ontology for the Dmoz topic Society.

2.2 Finding the Web Pages' Topics

One way to identify the topics of the pages that are available on the Web is to use the pages' classification provided by a Web Directory (such as Dmoz). However, considering that Dmoz (one of the largest Directories) classifies only 0.043%[e] of the pages that are visible to search engines, it becomes evident that we need to be able to determine the Web pages' topics regardless of whether they appear in a Directory or not. Therefore, we need to employ a classification scheme that categorizes Web pages into a set of predefined topics. Despite the plethora of the existing classification approaches (for a comparison of existing methods cf. [48]), in our work we chose to use a novel classifier, named TODE, which uses our topical ontology for estimating a suitable category to assign to every page. The main advantages that TODE classifier exhibits over the existing classification schemes summarize to the following:

- TODE is a powerful classifier that dynamically categorizes pages to the ontology's topics. Unlike, supervised classification techniques that need to undergo a rigorous training phase, TODE manages to automatically detect the topic of a page without any prior knowledge. Web pages' dynamic classification is of predominant importance for practical Web applications due to the constant modifications that Web data undergoes.
- TODE scales well in a dynamic environment where new data appears at prodigious rates. Based on a recent experimental study [38], we found that it takes TODE approximately 6 hours to categorize a set of 320,000 different Web pages into 13 Dmoz topics, including data cleaning and pre-processing time. For the same experiment, TODE demonstrated an overall classification accuracy of nearly 70% compared to the manual classification that the same pages exhibited in the Dmoz Directory. Note that for the same dataset a Bayesian classifier demonstrated an overall classification accuracy of 66%.

---

[e] While the size of the indexable Web is estimated to be nearly 11.5 billion pages [17], Dmoz contains roughly 5 million pages, as of April 2007.

The only shortcoming that TODE entails compared to other classification schemes is that it operates upon the availability of a richly encoded topical ontology, the categories of which will be used for annotating the pages' content. However, TODE's functionality is not restricted to a particular ontology and considering that there exist quite a few ontologies, we do not deem the ontology requirement as a significant impediment.

Given a topical ontology and a set of pages that one would like to categorize in the suitable ontology topics, we employ the TODE classification scheme and proceed as follows. We download Web pages, parse them to remove HTML markup, and we apply tokenization and POS-tagging. We then eliminate stop-words and we compute a set of indexing keywords for every page. Indexing keywords express the thematic content of the page and serve for finding the most suitable topic to assign to the contents of a given page. For keywords' selection, we did not to employ frequency-based techniques, (e.g., the TF.IDF weighting scheme), since we pursue the intuition that frequency alone is not a good indicator to drive the keywords' selection process further. This is because frequency information does not take into consideration the lexical cohesion of documents, i.e., the semantic relations that hold between the terms of a passage. Alternatively, we employ the lexical chaining method and we automatically generate sequences of thematic words for every Web page. The main stimulus for relying on lexical chains is that these provide an efficient technique for computing the lexical cohesion of a document. Formally, lexical chains are defined as groups or sequences of semantically related words. To compute the lexical chains of the pages' content, we adopt the approach described in [4]. Once we have computed lexical chains we disambiguate them on the basis of a scoring function f, introduced in [51]. In the following subsection we describe the basic steps of the above process. The reader familiar with lexical chains' generation and scoring may skip this subsection.

2.2.1 Computing Lexical Chains

Our intuition for using lexical chains to represent the contents of Web pages lies in the efficiency of the former in delivering for every page thematic words rather than merely keywords. Thematic words are informative of the pages' underlying themes ad can serves as a referential guide to the pages' contents, facilitating thus the categorization of pages into topics. Lexical chaining is a method for identifying indexing keywords for the Web pages and it is arguably less biased by the statistical distribution of terms within pages. Generally, the process for constructing lexical chains explores WordNet hierarchies and follows three steps (quoting from [4]): (i) select a set of candidate terms, (ii) for each candidate term, find an appropriate chain relying on a relatedness criterion among members of the chains, (iii) if it is found, insert the term in the chain and update accordingly. The relatedness factor in the second step is determined by the type of WordNet links that connect the candidate term and the other terms in a lexical chain.

We then disambiguate the words inside every generated chain by employing the scoring function f introduced in the work of [51], which indicates the possibility that a word relation is a correct one. Given two words, w_1 and w_2, their scoring function f via a relation r, depends on the words' association score, their depth in WordNet and their respective relation weight. The association score (*Assoc*) of the word pair (w_1, w_2) is determined by the words' co-occurrence frequency in a corpus and it is given by:

$$Assoc(w_1, w_2) = \frac{\log(p(w_1, w_2) + 1)}{N_s(w_1) * N_s(w_2)}$$

where $p(w_1,w_2)$ is the corpus co-occurrence probability of the word pair (w_1,w_2) and $N_s(w)$ is a normalization factor, which indicates the number of WordNet senses that a word w has. Given a word pair (w_1, w_2) their *DepthScore* expresses the words' position in WordNet hierarchy and is defined as:

$$\text{DepthScore}(w_1, w_2) = \text{Depth}(w_1) \bullet \text{Depth}(w_2)$$

where *Depth* (w) is the depth of word w in WordNet. Semantic relation weights (*RelationWeight*) have been experimentally fixed to 1 for reiteration, 0.2 for synonymy and hyper/hyponymy, 0.3 for antonymy, 0.4 for mero/holonymy and 0.005 for siblings. The scoring function f of w_1 and w_2 is defined as:

$$f_s(w_1, w_2, r) = Assoc\,(w_1, w_2) * DepthScore(w_1, w_2) * \mathrm{Re}\,lationWeight(r)$$

The score of lexical chain C_i that comprises w_1 and w_2, is calculated as the sum of the score of each relation r_j in C_i. Formally:

$$\text{Score}(C_i) = \sum_{r_j \text{ in } C_j} f_s\ (w_{j1}, w_{j2}, r_j)$$

To retain a single lexical chain for every downloaded Web document, we segment the document into paragraphs, identified by the HTML source tags. If a paragraph tag *<p>* is not found in the HTML source, we use the shingling technique, described in [52] where we group *n* adjacent words of a page to form an *n*-shingle. We define the size of $n = 50$, which roughly corresponds to the number of words in a typical paragraph. Therefore, we treat shingles as paragraphs of Web pages. For each paragraph, we generate lexical chains and we compute their scores using the formula described above. In case a paragraph produces multiple lexical chains, the chain of the highest score is regarded as the most representative paragraph's chain. By calculating the score of each paragraph's chain separately and retaining the chain of the highest score, chain ambiguities are eliminated. We then compare the overlap between the elements of each paragraph's lexical chain consecutively. Elements that are shared across chains are deleted so that lexical chains display no redundancy. The remaining elements are merged together into a single chain, representing the contents of the entire page, and a new Score ($C_{i'}$) for the resulting chain $C_{i'}$ is computed. This way we reassure that the overall score of every page's lexical chain is maximal. Moreover, via lexical chaining we narrow down every Web page to a sequence of thematic words, which subsequently reduces storage requirements. The elements of each chain are used as thematic keywords for assigning the underlying Web pages in topical categories.

2.2.2 Computing Topic Relevance Scores

In this section we present our approach towards assigning every web page to a suitable ontology topic and we describe how we can estimate the degree to which the identified topic is expressive of the page's content. To detect the topical category of a web page we rely on the page's thematic words represented in its lexical chain and we look them up in the ontology's concepts. Thematic words that match any of the ontology's concepts are further examined in order to derive their corresponding topical categories. Recall that every ontology concept is related (directly or indirectly) with a topical node representing a Dmoz, a SUMO or a MWND domain category.

Having annotated every keyword in a page with an appropriate ontology topic, the next step is to determine which of the keyword-matching topics is most appropriate for representing the page's entire content. To decide that, we introduce the *Topic Relevance Score* (*TRScore*) that computes how expressive is each of the ontology's keyword-matching topics in describing the content of a page. The *Topic Relevance Score* of a page p_i to the ontology's topic T_k is determined by the fraction of thematic keywords in p_i that are descendants (specializations) of T_k in the ontology, weighted by the Score of the page's chain. Formally, the *TRScore* of a page to each of the topics considered is given by:

$$\text{TRScore}_T(p) = \text{Score(Cp)}\frac{|\text{thematic words in p matching T}|}{|\text{thematic words in p}|}$$

where Score(Cp) represents the score of the lexical chain computed for page p and the denominator elimates any impact that the length of the chain might have on TRScore. Based on the above formula, we normalize the page-topic relevance (TRScore) values so that the latter range between 0 (meaning that the topic is not expressive at all of the page's content) to 1 (meaning that the topic is highly expressive of the page's content). Summarizing, relying on the above formula we compute the degree to which each of the ontology topics represent the pages' contents and based on the derived computations we assign every web page to all its matching topics accompanied by the respective TRScore values.

2.3 Web Pages' Topic Importance Scoring

To estimate how important every page is for the ontology topic in which it is classified, we employ the DirectoryRank (DR) metric, introduced in [30], which organizes the pages that are assigned to a topic based on the amount of information that every page communicates about the topic. DR defines the importance of a page p_i in a topic k to be the sum of its *Topic Relevance Score* (*TRScore*$_k$) and its overall similarity[f] to the fraction of pages n with which it correlates in the given topic, as given by:

$$\text{DR}_k(p_i) = \text{TRScore}_k(p_i) + \frac{1}{n}\sum_{k=1}^{n}\text{Sim}(p_i, p_n)$$

Intuitively, an informative page in a topic is a page that has a high relevance score to that topic and that is semantically similar to many other informative pages in the same topic. Based on the DR formula we can estimate the Web pages' importance for each of the ontology topics.

3 Automatic Identification of the User Search Preferences

We now discuss how we leverage the ontology to automatically identify a user's personal preferences based on her past click history. Our assumption here is that the topic preferences of a user are implicitly communicated via the queries she issues to search engines and exemplified by the topical categories of the pages that the user considers relevant for those queries. There are several factors that deter-

[f] For measuring the semantic similarity between two pages (p_1, p_2) we use the Dice coefficient which determines the similarity of p_1 and p_2 as: (*2*Common keywords*) / (*# of keywords in* p_1 + *# of keywords in* p_2)

mine what makes a page relevant for some query [1], such as the time spent on a page in conjunction to the page's length [13], the points of focus on a page [20], email, and/or bookmark of the page for future reference [42] and so forth. In our work, we deem a page to be relevant for a user with respect to some query if the user has spent a reasonable amount of time reading (i.e. being active) that page or if the user has bookmarked the page after reading it. Moreover, we consider the frequency with which a user (re)visits a page for a given query as an indicator of the page's relevance to that query. Having decided on what makes a page relevant to a query, we now turn our attention to how we can learn the topic preferences communicated in the user queries based on the topical analysis of the query-relevant pages in the user's past click data.

3.1 User Preference Identification

In this section, we built upon previous work [39] and we describe how our model identifies the preferences of a user based on the topical analysis of her click history. Our approach relies on the assumption that the user preferences hidden behind search queries can be represented at the level of the web pages that the user deemed relevant for those queries. In the remainder of the article we will refer to the query relevant pages as visited pages, where a visited page (in contrast to a viewed page) is a page the user has spent some time on reading it.

For identifying the user preferences, our model uses the topical ontology and the TODE classification scheme (cf. Section 2.2) in order to annotate every page in the user's click history with an appropriate topical category. It then groups the visited pages into topic clusters and also employs the DR formula (cf. Section 2.3) in order to rank the pages in every cluster according to their topic importance values. To regulate the effect of multiple page visits within a topical cluster, our model uses the logarithm of the number of page visits to weight their DR values.

Based on the visited pages topic clusters, we can easily produce a Query-Topic correlation lookup table with tuples in the form: $\langle p_1, DR_{T_i}(p_1) \rangle, \ldots, \langle p_x, DR_{T_i}(p_x) \rangle$ where x is the number of the pages visited for q that are classified in topic T_i and $DR_{T_i}(p_j)$ is the degree of importance for the j^{th} page in topic T_i. The data stored in the Query-Topic table indicates for a query q a set of candidate topics T for describing the search intention hidden behind q. Based on the above data, we compute the probability that q relates to each of the candidate topics T. Formally, we consider the probability that q relates to a topic T_i to be determined by the average DR values of the pages visited for q that are classified under T_i, as given by:

$$\text{Query Topic Relevance}(T_i \mid q) = \frac{\sum_{i=1}^{P_x} DR_{T_i}}{|P_x|}$$

where P_x is the number of pages visited for q that are assigned to topic T_i. Based on the above probability, we can compute for every query its degree of relatedness to each of the candidate topics. More specifically,we group the queries in the user's search history into topical clusters, with one cluster of queries per topic and based on the data in the query topic clusters, we can derive the user's degree of preference in each of the query topics as:

$$\text{QueryTopic Preference (T)} = \frac{\sum_{i=1}^{Q} \text{Query - Topic Relevance } (T \mid q_i)}{\mid Q \mid}$$

where q_i is the number of the user's past queries that relate to topic T and Q is the total number of past queries considered for that user. Based on the above formula, we estimate the user's degree of preference for some topic based on the topic relevance values of the user past queries, so that the greater the relevance between the queries and a topic, the stronger the user preference in that topic.

Although query-topic preference values are useful in determining the underlying topical intention of a given query, nevertheless these do not suffice for deriving the actual user profile. This is basically due to the fact that different users apply different search strategies and therefore they react differently to the search results that they are presented. In particular, a user might visit search results in response to some query and after a while conclude that these are not relevant to her query intention. In that case, casting the query intentions to the topics of the visited pages would result into misinterpretations of the user interests. In a different situation, a user might have a specific intention in mind when issuing a search query but upon visiting a number of search results this interest might change and be influenced by the visited pages' contents. In this scenario, the visited pages' topics might be reflective of the user's interest but these should not be attributed to the search query.

Considering the above, it becomes evident that there should be a balance between the impact that the topics expressed via the user queries and the topics discussed in the visited pages should have on the user profiling process. To address the above issues, our model considers not only the query topics but also the topics of the pages that are regularly visited by the user for particular query intentions (i.e. topics) across different searches. This is in order to ensure that our model will manage to capture the user interests even if they change during a single search or even if they are not adequately exemplified in the users' search requests.

To enable that, we rely on the topical categories of the user visited pages and we try to model the user's interest in the pages' topics as follows. Firstly, we employ the topic relevance values (TRScore) that each of the visited pages exhibits for each of the topics considered. Thereafter, we estimate the degree with which the topic of a page is preferred by the user across her searches as follows:

$$\text{Page Topic Preference (T)} = \frac{1}{|Px|} \sum_{P_i \in P}^{P} \text{TRScore}_T(P_i)$$

where Px is the total number of pages visited for all the user queries that relate to topic T and P is the set of pages visited by the user across her collected search trace. Based on the above formula we can derive the degree with which every topic in the previously visited pages was preferred by the user so that the greater the page topic preference values, the increased the probability that the page topic is reflective of the user's search profile.

So far we have presented the way in which our model computes a suitable topic to characterize the intention of a user's query as well as the way in which it manages to estimate the degree of the user's preference in the topics discussed in the user's search history. We now describe how we can put the above measurements together in order to derive the user's topical preferences. Formally, we estimate the degree of the user's preference in each of the topics identified in her search trace as the product of the user's interest in the topics estimated for her queries and the topics in the pages that the user has regularly visited across her past searches, given by:

$$\text{User Topic Preference}(T) = \text{Query Topic Preference}(T) \bullet \text{Page Topic Preference}(T)$$

Based on the above, we can derive the user's topical preferences without the need for user involvement. We now turn our attention to how we can utilize the learnt profiles in the query refinement process in order to suggest users personalized query alternatives.

4 Personalized Query Refinement

The challenge in realizing personalized query refinement is to enhance the user-issued query with alternative keywords that strongly correlate to both the initial query semantics and the preferences of the user issuing that query. To achieve this, we essentially need to disambiguate the user query and thereafter select a set of candidate terms to participate in the refined query. Query disambiguation practically translates into finding the user topic preference that is hidden behind a given query. In this section, we describe our approach towards identifying the topic of a new query for which there is no click data available (Section 4.1). We then, proceed with the description of how the identified query topic participates in the keyword selection process (Section 4.2). We conclude the section with a description of how the knowledge accumulated about the user preferences and the query topics is put together in order to formulate personalized refined queries (Section 4.3).

4.1. Query Topic Detection

The greatest challenge in every query refinement approach is to successfully resolve query sense ambiguities. Query sense disambiguation is vital in selecting semantically relevant terms to improve the user typed query. Considering that in our query refinement method we need the selection of alternative wordings take place not only in terms of the query semantics but also in terms of the user preferences, we realize that the challenge of our work lies in the effective detection of the topic preference that is hidden behind a user query. In this section, we study the query topic detection problem and we present our approach towards learning the preferences of a user's current query for which there is no click data available.

Our method relies on both the user past preferences and the query itself. In particular, consider that a user issues a query *q* to a search engine and retrieves a list of ranked results. Our objective is to identify the topic of *q* before the user clicks on any of the search results. To do so, upon query issuing, we firstly go to the user's search logs and we look for *q* among that user's past queries. If *q* is found, we take all the topics that have been previously identified for *q* along with their overall Query-Topic Preference values as the topics that are likely to represent the user intention latent behind the resubmission of *q*. Based on the observation of [5] that query trends remain relatively stable over time, we estimate the topic preference of a query resubmission as:

$$\text{Resubmitted Query Topic Preference}(T) = \frac{\sum_{i=1}^{T} \text{Query Topic Preference}(T_i)}{|T|}$$

where T is the total number of topics encountered in the user's search history, T_i is a topic attributed to a previous submission of *q* and Query Topic Preference (T_i) is the overall probability that the user preferred topic T_i in all her previous submissions of *q*.

On the other side of the spectrum, to detect the topic of a query that has not been previously submitted by a user, we employ a twofold approach. First we examine the semantic similarity between the current query and the queries previously issued by that user in order to determine whether the user's current query reflects a new topical preference or not. Our intuition here is that the more semantically similar the user queries are, the greatest the probability that they pertain to the same topical category and thus the more probable that this category reflects the user's topical preference.

To estimate the semantic similarity between the terms in a newly issued query and the terms in the search keywords previously submitted by the user, we employ the Wu and Palmer semantic similarity metric [45] and we use WordNet as our resource against which similarity will be measured. In particular, we map the terms of the current query as well as the terms in the previous queries of the user to their matching WordNet nodes and we derive the degree of similarity between pairs of words (w_1, w_2) as:

$$\text{Similarity}\ (w_1, w_2) = \frac{2 * \text{depth}\ \left(\text{LCS}\ (w_1, w_2)\right)}{\text{depth}\,(w_1) + \text{depth}\,(w_2)}$$

where LCS represents the least common subsumer (first common hypernym) of the two terms in WordNet hierarchy and depth(w) denotes the length of the path (i.e. number of nodes from the root) that leads to the WordNet matching node.

Based on the above formula, we compute the average paired similarity values between past and current query terms in order to determine the degree to which a new query represents a previously preferred topic. Formally, we firstly estimate the semantic similarity between a new query (q_{new}) and the past user queries (q_{old}) as:

$$\text{Query Similarity}\ (q_{\,new}, q_{\,old}) = \frac{1}{t}\sum_{i,j=1}^{t} \arg\max_{q_j} \text{Similarity}(q_i, q_j)$$

where t is the total number of terms in the past and new user queries. Since the appropriate sense for the new query is not known, our measure selects the senses which maximize Similarity (arg max similarity). Having estimated the semantic similarity between a user's new and each of her past queries, our next step is to determine an appropriate topic for expressing the new query's topical preference. Intuitively, one would expect that the increased the similarity between new and old queries, the increased the probability that a previously preferred topic can reflect the new query's intention. Based on the above intuition, we try to decipher the new query's intention by considering on the one hand the similarity values between the new and past user queries and on the other by accounting for the user preferences in the topics assigned to her previously issued queries. Recall here that by new query here we refer to a query submitted for the first time by the user.

For our estimations, we firstly group the past user queries by topic and we compute for every topic the overall similarity between the user's new and past queries for the topic. Formally, we derive the degree to which a newly issued query relates to any of the previously preferred topic as:

$$\text{New Query Topic Relevance}(T_i) = \frac{1}{|Q(T_i)|} \sum_{q_{old} \in Q(T_i)}^{Q(T_i)} \text{Query Similarity}(q_{new}, q_{old})$$

where Q(Ti) is the total number of queries previously issued by the user for expressing a user preference for topic Ti and Query Similarity (q_{new}, q_{old}) is the overall similarity between the new user query and the past user queries submitted for topic Ti.

Based on the above estimation, we can now approximate the probability with which a topic previously preferred by the user can reflect the user's interest that is hidden behind her new query as:

$$\text{New Query Topic Preference}(T) = \text{New Query Topic Relevance }(T) \bullet \text{User Topic Preference}(T)$$

where User Topic Preference (T) denotes the degree of the user interest in T as this has been estimated from the analysis of the user's past click history (cf. Section 3.1).

Based on the Resubmitted and New Query Topic Preference formulas presented in this section, we can compute the probability with which a topic previously preferred by the user is likely to reflect the user's current search interest, expressed via her present query (either previously submitted or not). Before we proceed with the description of how our personalized query refinement model operates upon the query topic detection, let's first summarize the basic steps of our query topic detection algorithm and discuss some practical considerations that arise form our approach. Figure 2 illustrates the pseudocode of our algorithm that operates upon the data collected from a user's past query trace and the topical preferences that have been computed for the user's past searches.

So far we have presented our method towards deciphering the query topic based on the analysis of the user's previous searches as well as on the semantic similarity between past and current user queries. The core of our method relies on the intuition that users retain a relatively stable number of interests in their web searches and that these interests can be determined from the analysis of their previous queries and search behavior. Of course one issue that our model cannot tackle instantly is when the user interests change drastically between consecutive search sessions and these interests are temporary in the sense that they are expressed via a small number of queries, which after a while disappear from the user's searching vocabulary. This might be the case when the user looks for information about a topic that was on the news one day and which falls beyond the user's general preferences. In such cases, we do not have enough evidence about the necessity to provide users with personalized query alternatives since we speculate that a user interested in some topic for a limited number of searches might not want that topic to characterize her general search profile that we are trying to build in the course of our present study.

Another issue we need to address is the flexibility of our method to incorporate different parameters or resources than the ones employed in our study. In this respect, we have designed our model to be as open-ended and customizable as possible but still it entails a number of pre-requisites. These concern the availability of a topical ontology, the availability of past user search logs as well as the exploitation of a semantic similarity metric. Given that there exist several topical ontologies and different semantic similarity metrics we believe that our approach could be fruitfully explored by others as this is not bound to any particular ontology or similarity measure. In the following section, we present our method towards selecting query term alternatives that are semantically relevant to both the user typed queries and the user preferred topics.

```
Take all queries Q in the user's search trace, grouped by topic
 // Find the topic preferences of a previously submitted query
  If current query q ∈ Q
        Compute for every T in Q the sum Query Topic Preference (T) over all topics
        in Q
        Return results and set them as Resubmitted Query Topic Preference (T)
        values for every T found
  Else
         // Find the topic of a newly issued query q
         Map terms in q and Q to WordNet
         For terms found compute Similarity (q terms, Q terms)
        If Similarity >0
                Compute Similarity (q, Q)
                For Similarity (q, Q1) > (q, Q2) > ....
                Keep maxSimilarity (q, Qi); then
                Group elements in Q by T and compute sum Similarity q,Q) for every
                T over all topics
                Return results and set them as New Query Topic Preference (T) for
                every T found
        Else
                Return unknown topic
```

Figure 2. Pseudo-code of the query topic detection algorithm.

4.2. Keywords Selection

So far we have presented our approach towards deriving the user topical preferences based on the analysis of her past clickthrough data (cf. Section 3.1) as well as our method for estimating the probability with which a previously preferred topic is reflective of the current query's intention (cf. Section 4.1). In this section, we discuss how we can combine the knowledge accumulated so far in order to assist information seekers improve their search queries and therefore experience successful web searches. To account for that, we designed a model that offers web users with alternative query wordings that are semantically relevant to their self-defined queries and at the same time expressive of the users' topical preferences.

The greatest challenge in the query reformulation process is to decide on the alternative keywords of a user typed query. In this direction, our model relies on two distinct yet complementary sources: the topical preferences of the user issuing the query and the query semantics. In the following paragraphs we describe how we can put these pieces of information together in order to derive a set of keywords that will be presented to the user for reformulating her query.

Given a query and a set of topical categories estimated for describing the preferences of the user issuing that query, our method selects alternative query keywords based on the following dual assumption: if the topics identified for a current query are highly preferred in the user's previous searches and if the keywords in the past user queries for the preferred topics are semantically similar to the keywords of the current query, then there is some probability that good query alternatives can be found within the user's previous searches. The intuition behind exploring the queries previously issued by a user for refining her current search is that a user might use different vocabulary for describing the same information need. In such case, it would be useful to supply the user with her past queries on the same search in order to formulate a precise and informative query.

Therefore, we define a threshold value H, being experimentally fixed to H = 0.7 which indicates whether the user's interest in the current query's topic is high or not. In case the user's New Query Topic Preference is above H we rely on the past user queries on the topic for deriving query alternative keywords. In particular, we take all past user queries submitted for that topical preference and we explore their Topic Relevance values (cf. Section 3.1, Query Topic Relevance formula). Form all the queries that have been previously issued for that user preferred topic, we select those whose Topic Relevance values fall within the top 10% of all the Query Relevance values for the topic. Selected queries are candidates for refining the user's current query. To pick from the candidate queries those that will be presented to the user as alternative query suggestions, we rely on their semantic similarity to the current query keywords. That is, we apply our Query Similarity formula (cf. Section 4.1) and we estimate how semantically similar is the user's current query to the past user queries. Thereafter, we sort past user queries by similarity values and we pick the first five most similar queries as alternatives for refining the user's current request.

On the other hand in case the topic of the current query has been marginally preferred (i.e. below H) in that user's previous searches, we select alternative query keywords by relying on the query semantics alone. As mentioned before, trying to personalize a query that discusses topics not generally preferred in the user's past searches might be questionable in the sense that these newly identified topics might reflect a temporary user interest and as such they might fade away soon from the user's search trace. Therefore, we would need a sufficient amount of user search logs before we can conclude on how persistent the user's interest is to some topics. Nevertheless, helping users refine their queries is a task that falls beyond personalization and it should be enabled regardless of the similarity between current and previous user searches.

In the course of our study, we improve the user's queries whose topics are less preferred by the user by relying on the query semantics and the query detected topics. In particular, we firstly map the query keywords to their corresponding ontology nodes and we compute the degree to which every keyword semantically relates to any of the query detected topics. For our estimations, we employ the semantic distance metric, introduced in [44], which derives the closeness between the concepts that match the query keywords and the concepts that match the query detected topics, given by:

$$\mathrm{dist}(c_1, c_2) = 2 \log(p(\mathrm{LCS}(c_1, c_2))) - (\log(p(c_1)) + \log(p(c_2)))$$

where LCS (c_1, c_2) is the least common subsumer of the two concepts (c_1, c_2) in the ontology and p(c) is the probability of encountering an instance of c_1 (query matching concept) or c_2 (query matching topic) in the local search results of the query. Based on the paired semantic distance values between the query and the query preferred topics, we derive the probability with which each of the identified topics is expressive of the query's intention.

Following on we rely on the query topic that is semantically closest (i.e. has shortest distance to) the query terms and proceed with the selection of alternative query terms as follows. We turn to the ontology and pick the synonyms of the query matching concept that is closest to a query topic as alternative query keywords. In case the query matching concept has no synonyms, we use the matching node's direct hyponyms (i.e. specialized concepts) as alternative query wordings. If however no hyponyms are found (i.e. the query matches a leaf node in the ontology) we rely on the matching node's siblings and we use them as keywords for refining that query. Overall, refining a query whose topic has not been generally preferred in the user's previous searches is mainly dependent upon the query semantics and the degree to which these are reflected in the identified query topics. Next we describe how refined queries are presented to the user and in Section 6 we give the details of an experimental evaluation we carried out and we discuss obtained results.

5 Visualizing Refined Queries

Unlike traditional query refinement techniques that offer a list of alternative words for improving a search request, our method uses graphical forms to visualize the improved queries and navigates the user to the desired information through the interaction with these forms. The intuition for organizing refined queries in graphical representations is that humans can better clarify their vague information needs if they look at the relationships between their queries and the system suggested alternatives. Figure 3 illustrates an example of a refined query graph generated for the information need that is latent in the user issued query *travel*.

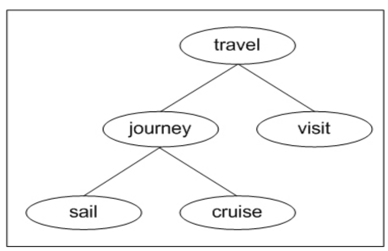

Figure 3. A refined query graph example

To assist Web information seekers define well-specified queries, we employ the keywords identified as the closest matches to both the user preferences and the query semantics (cf. Section 4.2) and we organize them in a graphical form as follows. We set at the root of the graph the topic identified for describing the current search intention of the user. Note here that among all the identified query topics, we employ those that are either highly preferred by the user throughout her searches or the one that is semantically closest to the query keywords in case no highly preferred topic (i.e. topic preference is below H) is found. We then proceed with the structuring of the children nodes and/or the leaves of the graph. Children nodes represent the system selected keywords (i.e. either past user queries or terms semantically related to the query keywords) and they are organized in terms of their conceptual relations encoded in the ontology. More specifically, having identified the ontology terms that are semantically related to the query keywords and which also relate greatly to the query detected topics we employ their ontology relations to both the query terms and to each other, in order to organize them in a graphical form. This way, we end up with a refined query that is visualized as a set of words (represented as nodes) that are interrelated through semantic relations (represented as links). Figure 4 illustrates by means of an example our approach towards building a refined query graph to improve the search of a user.

More specifically, consider that we need to refine a new query q_{15} issued by a user who has previously submitted a number of search queries. Assume now that we have found the topic preferences that are hidden behind that user's past searches (as discussed in Section 3) and let's say that these are represented by the topics: T_1, T_2 and T_3. Consider also that the most relevant queries for each of the identified topics are q_1, q_2, q_3, q_4 and q_5 (i.e. their Query-Topic relevance scores fall within the top 10% of all the values considered for the given topics) and they are distributed as follows: for $T_1\{q_1, q_2, q_3\}$, for $T_2\{q_4\}$ and for $T_3\{q_5, q_1\}$. Since q_{15} is a new query (i.e. not previously submitted by that user) we essentially need to detect the topic of q_{15}. For doing so we rely on the ontology and we compute the semantic similarity between q_{15} and every other query in the user's past search trace. Having estimated similarity values between q_{15} and each of the past user queries, we compute the degree to which q_{15} relates to any of T_1, T_2 and T_3 based on our New Query Topic relevance formula (cf. Section 4.1) and we proceed with the estimation of the user's preference for each of the above topics.

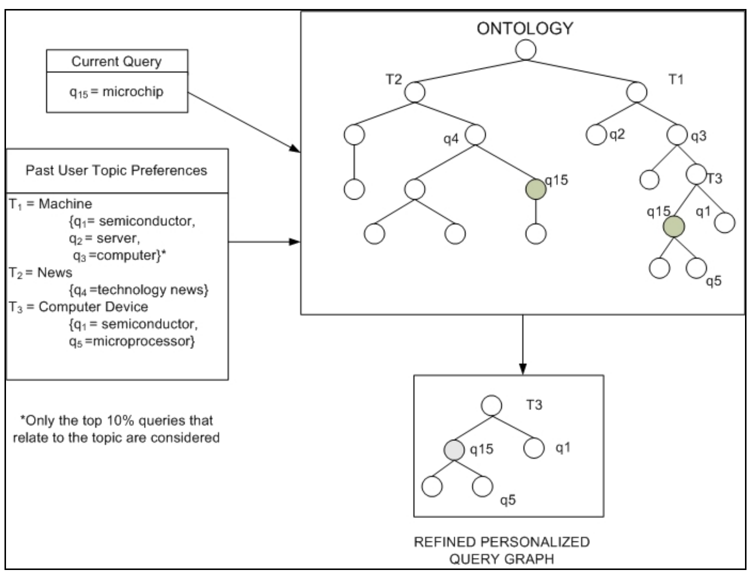

Figure 4. The refined query graph generation process

Assuming that the degree of the user's preference in T_1, T_2 and T_3 is below H (i.e. 0.7), we rely on the semantics of q_{15} for selecting alternative words for improving it. That is, we map q_{15} to the ontology and we estimate the semantic distance between the nodes that match q_{15} and the nodes representing T_1, T_2 and T_3. Note that a query matching multiple ontology nodes denotes a polysemous query. As

Figure 4 illustrates T_3 is the semantically closest topic of the three to the query matching nodes. Therefore, we pick all the hyponyms of q_{15} under T_3 and we present them to the user as query alternatives. We display the suggested refined query in a graphical representation following the organization that the query terms and the query alternative keywords exhibit in the ontology.

The user can interactively improve her initial query by clicking on any of the graph's nodes. The terms appearing on the nodes that the user clicks on constitute the keywords of the user's refined search. Clicking on nodes that have direct links to each other implies that the generated refined query is a Boolean 'and' query, whereas clicking on nodes that have no direct links to each other indicates that the generated refined query is a Boolean 'or' query. We should note here that visualizing refined queries in graphical forms facilitates the users' interaction with the system selected terms for improving their Web searchers and also promotes the users' searching skills by offering them alternative wordings to describe their information needs.

In this section, we have presented a novel approach towards providing Web users with improved queries in the pursuit of assisting people find the desired information when querying the Web. We believe that we have accomplished our endeavor by visualizing the refined queries in structures that are perceptible by humans and which resemble the way people organize knowledge in their minds. Next we evaluate the effectiveness of our approach in improving the retrieval performance and ultimately in ameliorating the users' search experiences.

6 Experiments

In this section we discuss the experiments we have conducted to evaluate the effectiveness of our personalized query refinement technique in improving the quality of retrieval results and therefore the users' web searches. We begin with the description of our experimental setup in Section 6.1. Then in Section 6.2, we describe the evaluation study we carried out in order to estimate the effectiveness of our approach in automatically learning the topic preferences hidden behind the user queries and we report on the obtained results. Finally, in Section 6.3 we experimentally compare the performance of our personalized query refinement technique in delivering qualitative and user-relevant results to the performance of other query improvement techniques and we discuss obtained results.

6.1. Experimental Setup

To evaluate how effective our personalized query refinement technique is in improving the search quality, we relied on the following data: (i) the clickthrough history of 15 users, (ii) the queries that our subjects issued and the set of pages that they deemed relevant to their queries and (iii) the topic-importance (DR) values of the pages to the topics in which they have been assigned. To collect these data, we have contacted 15 postgraduate students from our school and collected their search history for a period of two weeks. In particular, we recorded all the queries our subjects issued to Google during the experimental period and the search results that they clicked and stayed on (i.e. they were active) for more than 10 seconds. Ignoring queries with no results, or where no results were visited for more that 10 seconds, we collected a total of 1,066 queries. Of those, 843 were overlapping queries, i.e. submitted more that once by our subjects, and 223 were unique queries, submitted only once by our users. On average, every participant issued 71.07 queries of which 14.47 were issued only once and 56.60 were submitted multiple times.

To identify the set of topics that are hidden behind our collected query trace, we carried out a user survey, where we asked our participants to keep a diary of their web searches during their participation in the study, and indicate for every query right after the query submission the topic preference they had in mind. Topic preferences were indicated in two manners, at the participants' discretion. In one approach, users were asked to choose a topic to describe a web search based on a list of hierarchically structured topics that we provided them. These topics are the text descriptors of the categories in the first 2 levels of the Dmoz Directory that are compatible with the topics in our ontology. Of the 619 categories in the first two levels of the Dmoz Directory, 172 are represented in our ontology. In another approach, users were asked to define their topic preferences using their self-selected category descriptors in case none of the ones provided to them could adequately capture their search intention. On average each of our subjects was interested in 5.4 different topics. Of these, 5.12 were ontology topics and 0.28 self-selected topics. Note that our subjects were not given any details about the nature or the scope of our study. Moreover, once they indicated their topic preferences they could not go back and alter their selections. Finally, we asked our participants to report their searches (e.g. issued queries, preferred topics and relevant pages) on a daily basis. Following submission of a user's search report, the latter was made inaccessible to the user, in order not to influence her future judgments on topic preferences. All participants were postgraduate computer science students, with high levels of computer literacy and familiarity with web search.

To identify the set of pages that our subjects deemed the most relevant to each of their preferences, we asked them to select among the set of pages that they have visited for a query, the pages that they considered as the best matches for their information needs hidden behind each of their queries. Note here that user relevance judgments were made on the visited pages that were directly reachable from the search results without considering pages reachable after following more than one links from the retrieved results. On average 4.1% of the visited pages were considered relevant to each search intention by each subject in our survey. Table 1 summarizes some statistics on our experimental data.

Table 1. Statistics on the experimental dataset

Collection period	April 15-30 2007
Number of users	15
Number of queries	1,066
Number of visited pages	7,441
Avg. # of queries/user	71.07
Avg. # of topic preferences/ user	5.4
Avg. # of visited pages/query	6.98
Avg. # of alternative keywords/query	3.8
Avg. # of nodes/refined query graph	4.1

To compute the topic importance values (i.e. DR) of our experimental pages we worked as follows. We downloaded a total set of 648,629 web pages which are listed under the Dmoz topics that are also represented in our ontology. Recall that 172 Dmoz topical categories are among our ontology topics. Having collected a small web corpus for each of those 172 topics, we assigned each of our experimental pages to a suitable topic based on the human judgments for the pages' topics. That is, depending on the topical category for which our participants evaluated a given page as relevant, we decided the topic in which our experimental page was assigned. Based on the above web collection we computed the DR (cf. Section 2.3) of our experimental pages to each of their corresponding topics.

Moreover to estimate the topic importance values of the pages judged as relevant for the user-defined topics, we used the text descriptors of the user selected topics as queries, which we submitted to Google search engine. We then downloaded the first 200 pages retrieved for each those queries and we computed the importance (DR) values of these pages to their respective topical categories, as the latter were indicated by our subjects. Table 2 shows the overall distribution of the experimental pages in the generic ontology topics and reports the number of subtopics concerned for each of the top level categories. In our evaluation we rely on the page's specific topics as these were determined by our study participants and we do not restrict our assessment to generic categories. Note also that the *misc* category in Table 2 represents the topical categories defined by the users and which are not represented in the ontology.

In sum, our experimental dataset comprises of: (i) the set of queries that 15 users issued to Google for a period of two weeks as well as the set of pages visited for each of those queries by each of our subjects, (ii) the topic preference for each query as this is given explicitly by the person who issued that query, (iii) the set of pages that our subjects considered relevant to their self-selected queries, and (iv) the topic importance values (DR scores) of nearly 650,000 web pages (including user visited pages) that are listed under the topics that our subjects selected for indicating their query-topic preferences. In the following paragraphs, we describe how we used our experimental data to evaluate on the one hand the accuracy of our model in learning the user topic preferences automatically and on the other the effectiveness of our personalized query refinement method in improving the quality of search results compared to the effectiveness of other retrieval improvement techniques.

Table 2. Topic distribution in our dataset

Category	**% of pages**	**# of sub-topics**	**Category**	**% of pages**	**# of sub-topics**
Arts	10.35%	16	Regional	2.22%	4
Games	7.14%	20	Society	8.38%	14
Kids and teens	2.46%	4	Computers	7.53%	13
Reference	3.50%	10	Home	5.19%	7
Shopping	9.28%	15	Recreation	5.83%	18
Business	7.46%	7	Science	9.38%	9
Health	4.30%	6	Sports	7.98%	19
News	8.27%	5	misc	0.73%	5

6.2. Accuracy of the User Profile

The best way to evaluate the learning accuracy of our user profiling method is to directly measure the difference between the users' actual topic preferences and the topic preferences that our model estimates based on the analysis of the users' past clickthrough data. For our evaluation, we used the topical preferences that our subjects have explicitly identified for each of their issued queries and the set of pages that they have indicated as relevant to their search pursuits in order to derive the degree of the users' actual topical preferences. More specifically, given a number of queries that a user has issued, the topical descriptors that the user has selected for describing her query intentions and the set of pages that the user has evaluated as relevant to her search pursuits, we compute that user's actual topic preferences as follows.

We firstly, apply our Query Topic Preference formula (cf. Section 3.1) in order to compute how much each of the user queries relates to the topical category indicated by the user as query descriptive. Thereafter, we employ the TRScores of the pages that the user considered relevant for each of her queries in order to derive the Page Topic Preference values for that user. Finally, we apply our User Topic Preference formula which combines the Query Topic Preference values for each of the user issued queries and the Page Topic Preference values for each of the user relevant pages in order to compute the degree of the user's interest in each of the topics considered in her search trace. Based on the estimations of our User Topic Preference metric (cf. Section 3.1) we derive the user's actual topic preferences, which we denote as P_A. Note here that since the query topics are explicitly determined by our participants, we deem the user's interest that our metric computes as actual user preferences. Note also that we estimated the degree of the users' preferences in their self-defined topics instead of asking our subjects to rate their preferences, because it tends to be difficult for users to assign an accurate weight to each of the selected topics.

Having computed the degree of user preferences to the topics that they have actually selected for describing their query intentions, we proceed with the estimation of the user preferences that our model delivers for the same set of pages and queries. In particular, we apply our User Topic Preference formula in order to derive the users' profile based on the analysis of the user's search logs alone and without considering anything about the query topics or the query relevant pages that our subjects have previously determined. In other words, we evaluate our model's accuracy in learning both the query and the page topic preferences of the user and therefore in computing an accurate user profile.

In this respect, we rely on the set of pages visited for each of the user queries, which we categorized in the ontology topics and we compute their TRScore and DR values. Based on the above pages, we estimate the Query Topic Relevance values and then we derive the Query Topic Preference scores following the steps presented in Section 3.1. Based on the query visited pages and their topic relevance scores (TRScores) we estimate the Page Topic Preferences which we subsequently combine to the Query Topic Preference values in order to estimate the degree of the User Preference in each of the topics considered. Learnt user preferences are denoted as P_L and they represent the user's interest in the ontology topics as these have been automatically computed by our formula and without any user involvement.

Based on the above data, we came down to two sets of user preferred topics (actual and learnt ones) where every topic is given a value (between 0 and 1) computed by our User Topic Preference formula according to how much the user has preferred that topic in the set of queries and pages considered for that user. The first set of topical preferences contains the topics that our subjects indicated as interesting for their issued queries and visited pages (denoted as P_A), whereas the second set of topical preferences contains the topics that our model estimated as interesting for the same set of user queries and visited paged (denoted as P_L). For our evaluation we sorted the topics in both lists in terms of their User Topic Preference values so that the most preferred topics (either actual or learnt ones) show up first in the respective set. This way we created for each of our study participants two sets of topical preferences; the actual and the learnt ones, which we compared in order to assess our model's accuracy.

For our comparisons, we compared the top-k elements in the sorted sets of actual and learnt topical preferences, using the Kendall's distance metric [22]. Formally, the Kendall's distance metric (τ) between two ordered sets indicates the degree to which the relative ordering of the topical preferences in the two sets is in agreement and is given by:

$$\tau\,(P_L, P_A) = \frac{\left|(\,i, j):i, j \in R\,, P_L(i) < P_L(j),\ P_A(i) > P_A(j)\right|}{|R| \bullet |R - 1|}$$

where P_L denotes the ordered set of the top-k preferred topics learnt by our model and P_A denotes the ordered set of the top-k preferred topics computed from the user's actual topical preferences, R is the union of P_L and P_A and (i, j) is a random pair of distinct topics. τ values range between 0 and 1, taking 0 when the two orderings are identical. Given that most users in our study were interested on average in 5.4 distinct topics, we set the value of k between 1 and 12. We believe that the choice of k is reasonable based also on the observation that none of our subjects indicated an interest in more than 10 different topics in their examined search trace[g].

Figure 5 illustrates the agreement between the actual and the learnt user preference for every study participant. The figure plots in logarithmic scale the degree of agreement between the ranked sets of actual and learnt user preferences for the number of topics considered for each of our users. Note that lower τ values indicate a stronger agreement between the sets' paired orderings.

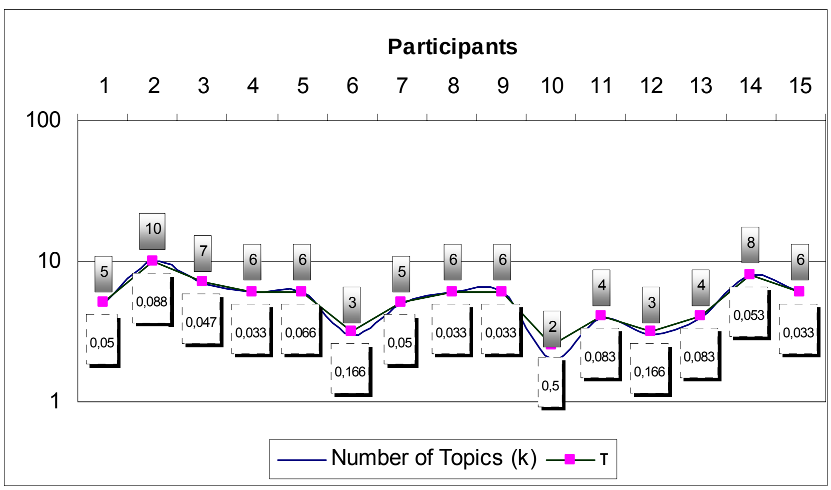

Figure 5. Ordering agreement between the actual and learnt topic preferences (lower is better)

Obtained results demonstrate that our method has a significant potential in automatically identifying the topical preferences hidden behind user queries. Note that our evaluation is not restricted to top level ontology topics but rather it concerns specific topics that our subjects have indicated for describ-

[g] An interesting observation is that although our participants were supplied with a list of 172 distinct topics to denote their preferences, these were limited to maximum 10 topics. This implies either that our subjects were reluctant to browse the list of all possible topics, or that they were interested in a relatively small number of topics, or a combination of the two.

ing their queries and which our model has also computed for building the user's search profile. In particular, our method has an overall accuracy of 0.098 in learning the user preferences, which practically translates to 90.2% successful performance in automatically deriving the user interests based on the analysis of past user queries and visited pages.

An interesting observation is that when the size of k is too small or too big, the learning accuracy of our model generally decreases. This implies that if the user is interested in only a small number of topics the probability that our model will capture and order them in a way the user would, is reduced. This can be attested in the results obtained for user 10, who indicated only two preferred topics in her examined query trace. The topics that our model computed for user 10 are three and even though both user selected topics are among the topics learnt by our model nevertheless their ordering (i.e. degree of preference) is not in agreement between the two sets. Therefore, the learning accuracy of our model for building the profile of user 10 is quite low. Likewise, if the user is interested in a large number of topics (i.e. more than 7) the probability that our model will compute the exact same topics for representing the user's profile becomes low. This trend is reasonable because when k becomes large, the user issues queries on diverse topics and as such we need a significant amount of clickthrough data to distinguish between the different topics. In general, results demonstrate that when the topic variance in the users search trace is moderate (i.e. between 3 and 7 topics) our model's effectiveness not only in identifying these interests but also in ordering them according to the user preference is quite high. Considering that web users are generally interested in different 5 topics on average, we might conclude that our model can be effectively employed for building user search profiles. To further validate the results obtained so far, we took a second evaluation approach presented next.

6.2.1. Meaningfulness of the User Profile

Despite the Kendall's distance metric efficiency in evaluating how well our user profiling method can compute the user preferences for a fixed number of topics (i.e. value of k) it is not sufficient for capturing the semantic correlation between the actual and the learnt user preferences when these are of varying size and/or quality. In particular, considering that we encouraged our participants determine their own topical preferences, beyond the 172 topics that we showed them, and given that the number of learnt preferences for a user might be greater or smaller than her actual preferences, it becomes evident that we need to measure the semantic similarity between the actual and the learnt user preferences before we conclude on our method's effectiveness in building meaningful user search profiles.

Before delving into the details of our second evaluation approach, we should underline that another motive of this assessment lies on the observation that although our subjects were supplied with a long list of topics (i.e. 172 topics) to denote their preferences and they were generally interested in a small number of topics (i.e. less than 11), nevertheless in some cases they defined their own topics to describe their preferences. Therefore, it is interested to see whether their self-selected topics are indeed new topics (i.e. absent from the ontology) and as such they should be included in the ontology, or whether these are very much similar to the topics that we provided them. If the second hypothesis is verified, it will imply either the users' difficulty in comprehending the semantics of the ontology's categories, or the ontology's limitation in offering meaningful categories within its contents.

To address the above issues we carried out a second evaluation where we measured the semantic similarity between the topics selected by our subjects for describing their search preferences and the topics estimated by our model for representing the user preferences. The motive for our second evaluation is to investigate the cases that the estimated user profiles are different from the actual ones and measure the difference between the two. To compute the similarity between the actual and the learnt user preferences we rely on the ontology and we estimate how similar is an actually user preferred topic (denoted as T_a) to a learnt user topic (denoted as T_l) based on the number of common subsumers (i.e. generalizations) between the two, given by:

$$S(T_\alpha, T_l) = \frac{2 \bullet \left| \text{common subsumers of } T_a \text{ and } T_l \right|}{\left| \text{subsumers of } T_a \right| + \left| \text{subsumers of } T_l \right|}$$

Based on the above formula we estimate the similarity between pairs of actual and learnt user topics so that the greater the similarity between the two, the better our model's accuracy in computing meaningful user profiles. In Figure 6 we present the results of our evaluation on the topic similarities between the actual and the learnt user profiles. The scores reported in Figure 6 are the normalized similarity (S) values for our experimental queries, averaged by users and averaged across the topics considered for each of our participants. The figure illustrates the average similarity values between the topics defined by each of our participants and the topics that our system learnt for every participant. As we can see in Figure 6 there is a considerable similarity between the actual and the learnt user preferences even if there is variation in the vocabulary used to describe them. In particular, we observe that the topic preferences estimated by our method over all our user queries yield an average similarity of 0.824 to the topics actually intended in our user queries. This practically means that even if our model fails to identify a user selected topic for describing that user's profile, nevertheless it still manages to find a topic very similar to the one indicated by the user. Therefore, we may conclude that our model has a significant potential in learning the user topical preferences.

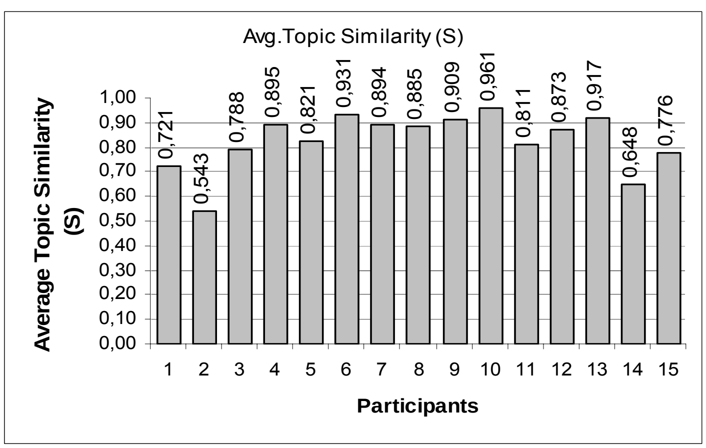

Figure 6. Average normalized similarity values between actual and learnt preferences

An interesting observation from the results reported so far is that although our model has a low accuracy in learning the topical preferences for users 10, 6 and 12 who were interested in a small number of topics, nevertheless the similarity between the learnt and actually preferred topics for those users is quite high. This implies that our model managed to identify a correct topic for building these user profiles but it was not that successful in ordering the learnt preferences in the same way that the respective users did. Consider for instance the case of user 6 who indicated the following topical preferences for her examined queries: Computer Graphics, Computer Animation and Film Festivals with that order of preference. The topics that our model estimated for user 6 are Computer Animation, Computer Graphics, Film Noir and Film Festivals (with that order of preference). Comparing the relative ordering between the two sets of preferences results into a low agreement; but a closer look at their paired semantic similarity values reveals that there is a strong correlation between the actual and learnt user preferences. As a matter of fact, all of the user selected topics are among the topics that our model estimated for user 6 but their ordering (i.e. indicating the degree of user preference) is quite different. The same applies to the topical preferences for users 10 and 16. Therefore, we might conclude that our model manages to accurately capture the user topical preferences even if these pertain to a limited number of topics, regardless of the way in which preferences are ordered.

On a different setting now, we wanted to examine our model's effectiveness in capturing the user interests when these span multiple topics (i.e. more than 7). As we mentioned above when the user interests increase, our model's accuracy generally decreases. This in attested in the results that Figure 5 reports for users 2 and 14 who are interested in large number of topics. In our previous evaluation we observed that there is relatively low ordering agreement between the actual and the estimated preferences for these users. In the present evaluation we seek to investigate whether the above disagreement is due to topic or due to ordering discrepancies in the two sets. As we can see in the results of Figure 6, the topics computed for users 2 and 14 are not very similar to the topics selected by these users for describing their preferences. As such we might conclude that our model is not very successful in deriving accurate profiles for users whose interests span several topics.

Another interesting observation is that from all our subjects, only one (i.e. user 2) defined her topic preferences using mainly self-selected descriptors, while the majority of our participants (i.e. 8 out of 15) defined their preferences based on the ontology descriptors alone. A further examination of the above results indicates that the topic preferences for user 2 were overly specific (e.g. tuple relational calculus, free order languages) and being such they were not represented in the ontology. Moreover, we observe that users 1, 3, 14 and 15 although they generally relied on the ontology topics for denoting their preferences, in some cases they defined their own topics as well. However, these self-defined topics exhibited some degree of relatedness (e.g. similarity) to ontology topics (cf. the average similarity between the topics defined and the topics learnt for user 3 is 0.788). This implies that our subjects might have been overwhelmed by the amount of topics that we showed them and as such they preferred to determine their preferences using their own terminology rather than go over a long list of topics before selecting the most suitable one. From a different perspective though, it might imply that our participants had a crystal clear idea about their search intentions and therefore they preferred to define their topics using a vocabulary they could control rather than having to distinguish between the numerous topics presented to them. Whatever the case, the above observations merit further investigation from a societal point of view, which goes way beyond the present study. In overall, experimental results demonstrate that our method is quite effective in approximating the users' search intentions even if it does not always pick the exact same topic that a user would define for representing her topical preferences and even if it does not always order the estimated topics in the way users would.

The results obtained so far indicate that our approach has a significant accuracy in learning a suitable topic to describe the user interests without any user effort. Another interesting finding is that the topics learnt from our system are the most relevant to the actual topics selected by users, when selection is based on the ontology's concepts. This demonstrates the usefulness of our ontology in capturing the latent user interests in an automated yet effective manner. Finally, a combination of the results shown in Figures 5 and 6 indicates that the more focused user preferences are to a small number of topics, ($1 \leq k \leq 6$) the more effective is our method in capturing these preferences automatically. Moreover, results imply that even when the user preferences are very diverse (e.g. the case of users 2 and 14), our profiling method that relies on the ontology, can compute a meaningful topic that generally suits the true intention of the user searches. The results reported so far provide strong indications of our method's efficiency in accurately learning the user preferences automatically. However, before we proceed to the investigation of how learnt preferences contribute to the improvement of the user searchers, we experimentally study the users' ability in submitting informative queries that discriminate their varying information needs.

6.2.2. Topic Variance in the User Preferences

Having obtained experimental evidence on our method's accuracy in detecting the user preferences hidden behind search queries, we now proceed with the evaluation of how informative search queries are about the users' true intentions. In particular, we investigate how humans attribute topical categories to their queries in an attempt to gain some insight on the users' behavior when querying the web.

What motivated this study is the observation that some of our participants although they issued the same query more than once, and they generally visited the same results for that query, nevertheless they selected different topics to describe their intentions across query submissions. To realize our pursuit, consider the following situation. User *A* issues query *q* twice, with a few days interval between submissions. During submission the user was asked to indicate the topic preference of her query, using any of the approaches discussed in Section 6.1. Note that our subjects do not have access to neither the queries nor the topics of their previous days' searches. Consider now that in both submissions of *q*, user *A* visited more or less the same set of pages and considered *N* of them as the best matches for her query. However, the topics the user selected for describing the query intention were different between the two submissions of *q*.

To understand the way in which users describe their search intentions, we collected from our experimental data all the queries that have been submitted more that once by the same user and which have been annotated with multiple topics by the user. Out of the 843 overlapping queries in our experimental data (cf. Section 6.1); 89 were multiple submissions, in the sense that they have been issued by the same user more than once. Moreover, 58 of these multiple query submissions have been annotated by our participants with more than one topics and for 24 of them, topic selection was determined from the same set of pages that the user deemed relevant to her respective query. Table 3 reports the distribution of the multiply submitted queries in our participants and indicates the number of topics that our users have attributed to these queries.

As we can see in Table 3, users 1, 13 and 14 who issued a number of queries multiple times during the recorded period, they indicated different topical preferences across query submissions although they generally relied on the same set of pages for casting their preferences. Therefore, it is interesting to see if the different topics indicated for a given query describe distinct user interests or if they all describe similar search intentions. If the former scenario applies (i.e. same query and pages but distinct intentions) this might imply that users have difficulty in formulating informative queries. Conversely, if the second scenario applies (i.e. same query and pages, different but relevant topics) this might imply that users crystallize after a number of queries their search pursuits and they are better able to explicitly express their intentions via ontology topics.

Table 3. Statistics on multiple query submissions

Users	**# of queries with multiple submissions**	**# of topics assigned**	**Same relevant pages**
U_1	6	2	YES
U_2	13	4	NO
U_3	6	1	
U_4	7	2	NO
U_5	4	1	
U_6	2	1	
U_7	8	2	NO
U_8	6	2	NO
U_9	5	1	
U_{10}	1	1	
U_{11}	2	1	
U_{12}	3	1	
U_{13}	8	2	YES
U_{14}	10	3	YES
U_{15}	8	1	
Total	**89**		

To examine which of the two scenarios applies to our three study participants, we decided to examine the semantic correlation between the different topics indicated for the same query across its submissions. In this respect, we relied on the ontology and we employed our query topic similarity (S) formula, described in Section 6.2.1 in order to estimate whether the different topics attributed to a given query pertain to a similar user interest or not. More specifically, given a query and a set of topical categories indicated for that query we measured the semantic similarity between pairs of query topics. Figure 7 reports the overall similarity values between the different topics that have been selected for describing the same search intention.

In the figure the first six queries represent the searches of user 1, the next eight queries represent the searches of user 13 and the last ten queries represent the searches of user 14. As we can see in most cases the different topics attributed to those 24 multiply issued queries are semantically close to each other, which practically means that although the text descriptors selected for representing the query intentions may vary, nevertheless they all pertain to the same domain area, i.e. they are all closely related topics. For instance consider query 6 (q6 = quake), which has been assigned to topics: Games and Computer Games, by user 1 based on almost the same set of visited pages across the query submissions. The semantic similarity in the ontology of the two topics amounts to 0.947 and a closer look in the ontology reveals that Computer Games is a hyponym of Games. Based on the above, we conclude that both query topics express more or less the same user interest and as such varying query topics are due to the user's attempt to specify better queries during her web searches.

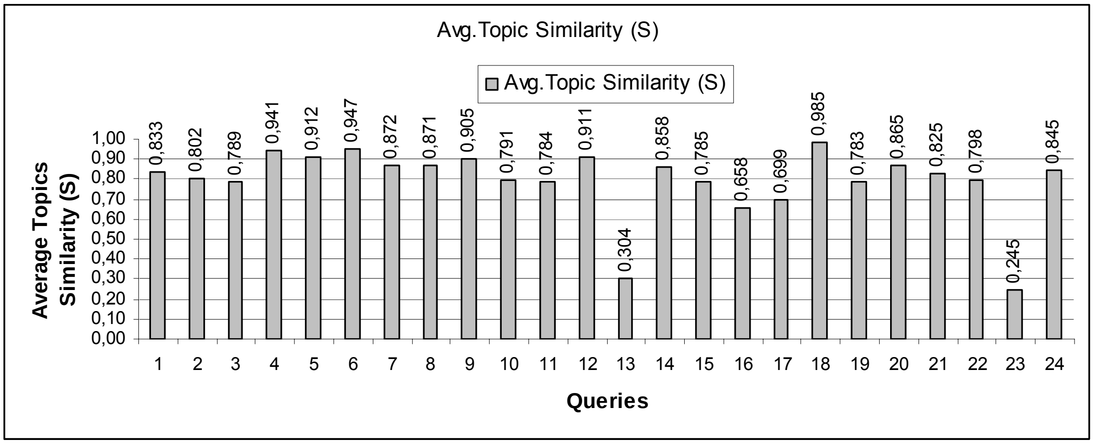

Figure 7. Average normalized similarity values between the different query selected topics

On the other hand, results demonstrate that for queries 13 and 23 the different topics that have been selected for representing the user intention are semantically dissimilar, i.e. they correspond to loosely related ontology concepts. Therefore, we may speculate for those cases that either the users had not a clear search pursuit when issuing those queries or they had difficulty in distinguishing between the ontology topics. Consider for instance query 23 (q23 = best price for Nokia T91), which has been assigned to topics: Cell Phones, Technology Financials and e-Shops, by user 14 based on almost the same set of visited pages across the query submissions. The overall semantic similarity in the ontology between pairs of the above topics amounts to 0.245. Based on the above, we may conclude that these multiple query topics denote a vague user interest that could not be precisely expressed via a single ontology topic.

Overall, results indicate that web users may select different topics to articulate their search intentions nevertheless in most cases these topics correspond to close matching categories. In an attempt to interpret our findings we speculate that although users may narrow down or broaden their search interests across repeated searches, nevertheless they have a stable generic notion of the information they wish to obtain for each of their queries. This is attested in our findings, which show that the majority of the topical categories attributed to a query throughout its submissions are semantically similar to each other. To validate the soundness of our assumption, we turn our investigation to how refined queries can assist users gain more control over their searches. The contribution of our personalized query refinement technique on retrieval performance is discussed next.

6.3. Search Quality of Personalized Query Refinement

To evaluate the effectiveness of our personalized query refinement technique in improving the quality of search results, we worked as follows. We relied on the recorded search history of our 15 study participants and we picked for each of our subjects a set of 10 random queries from their search trace. We then applied our personalized query refinement technique to those 150 queries and we compared our method's performance to the performance of other well-known query improvement techniques. The set of 150 experimental queries that we used in the present study correspond to different query types, such as informational, navigational, single term, multi-term and ambiguous queries. For our evaluation, we employed three different techniques for refining our 150 collected queries and we compared the quality of the results returned for our queries by each of the techniques considered. In particular, we compared the following query refinement methods.

LOCAL REFINEMENT: Given a query *q* and a ranked list of results returned for *q*, we employ the first ten pages in the results of *q*, we pre-process them in order to extract from their contents a set of keywords for reformulating the query. In particular, pages' pre-processing accounts to HTML parsing, tokenization, POS-tagging, stop word removal and lemmatization. Thereafter, we weight the pages' lemmatized terms by applying the TF.IDF weighting scheme and we retain *n* (*n*=25%) most highly weighted keywords as the basis for disambiguating the query. For disambiguation, we map the pages' keywords together with the query terms to the ontology's nodes and we pick the sense that maximizes its closeness to the majority of the pages' keywords, as the sense of the query. We then employ the synonyms of the selected query sense as the alternative wordings for refining the query. In case the ontology's concept that matches the query has no synonyms, we rely on the query's hyponyms for picking alternative query keywords. This refinement technique relies only on the query semantics and does not take the user preferences into account. We will refer to this method as LOCAL.

ANCHOR TEXT REFINEMENT: Given a query and a set of query matching pages, we rely on the anchor text of the pages' that point to the results of the query in order to extract alternative query keywords. In particular, given a set of pages that point to the search results of the query, we define an anchor window of 20 terms around a query term and we employ the query surrounding nouns as alternatives for query formulations. This refinement technique, proposed by [26] does not account for the query semantics neither for the user preferences in the query refinement process. Rather it employs terms appearing in the clickable text that is displayed for a hyperlink in an HTML page that points to a result retrieved for the query. We will refer to this refinement method as ANCHOR.

PERSONALIZED REFINEMENT: Given a query and a number of topics that represent the profile of the user issuing that query, we follow the steps described in Section 4 in order to firstly detect the user preferences latent behind the current query. Thereafter, we explore the identified topics for the query and the query semantics in order to select alternative query keywords either from the user's past searches on a relevant topic or from the ontology terms that are semantically similar to the query terms. This query refinement method uses both the learnt user preferences and the query semantics to identify alternative query wordings. We will refer to this method as PERSONALIZED.

To measure the retrieval performance of the above techniques, we relied on our 150 experimental queries which we reformulated in three different manners. In the first reformulation, the queries were improved with alternative wordings that were computed based on the local results of the query (i.e. LOCAL). In the second reformulation, the queries were improved with alternative wordings that were computed based on the anchor text of the query-matching pages (i.e. ANCHOR). Finally, in the third reformulation, the queries were improved with alternative wordings that were determined based on the query semantics as well as the query relevance to the past user topical preferences (i.e. PERSONALIZED.

Having applied each of the refinement techniques to our experimental queries, we presented them to our subjects and asked them to resubmit their requests by picking alternative terms suggested by each of the three refinement methods. In particular, we asked our subjects to re-submit each of their 10 examined queries three different times. In every submission we asked them to select alternative query wordings from a different refinement technique. Note that our participants were not given any information about the way in which alternative queries were determined, nor they were given any instructions as to which of the suggested terms to use in each of their query resubmissions. Moreover, considering that LOCAL and ANCHOR refinement methods rely exclusively on the query terms for selecting alternative keywords, in our evaluation we showed to our participants only the initial query terms and their alternative keywords that each of the methods selected, without conveying any information about the query topics. The only instruction our subjects were given was to indicate a single query alternative for every refinement method, by selecting the alternative that they considered as the most suitable for substituting each of their examined respective queries. Our evaluation relied on a total set of 450 queries that our subjects issued to our corpus of experimental web pages. Every participant issued 30 queries grouped in 10 different intentions, one for each of their 10 randomly selected queries. For every intention, our subjects defined the vocabulary of their queries in three different ways; based on the terms suggested by (i) the LOCAL refinement technique, (ii) the ANCHOR text refinement technique, and (iii) the PERSONALIZED refinement technique.

At this point we should note that by query intention we do not imply the topical preference that is latent behind a given query but rather the fact that a single query generally intends the retrieval of information about a single topic. When it comes to query detected topics these are determined as follows. For the queries submitted under the LOCAL refinement technique, their topical categories are determined by the topics that have been assigned to the identified query synonyms. Likewise, for queries submitted under the ANCHOR text technique, their topical categories are determined by the topics that have been assigned to the anchor text keywords of the query matching pages. Finally, for queries submitted under the PERSONALIZED refinement technique, their topical categories are determined based on the queries' semantics and their relevance to the user preferred topics as discussed in Section 4. Recall though that query topics were not displayed to our subjects in order not to influence their judgments about keywords' selection. Following the resubmission of the experimental queries, we explored the first twenty pages retrieved for every query in order to evaluate the retrieval performance of the query refinement techniques examined. Note that queries were submitted against the dataset of roughly 650K pages that we collected for our experiments (cf. Section 6.1) and the ranking of the pages retrieved for each of the queries was determined by the DR formula, which orders pages in terms of their importance to the topics (i.e. semantics) that have been assigned to the respective queries.

To evaluate the effectiveness of each of the three query improvement techniques in retrieving qualitative results, we relied on the precision of the first twenty pages retrieved for a query in each of the query's refinement submissions. In our evaluation, we determine precision as the fraction of pages within the first twenty results of a query that have been identified by our participants as the best query-matching pages. Recall that while recruiting our users, we asked them to indicate for each of their queries which of the retrieved pages they considered the best matches for the respective queries. Formally, we define the precision of the results returned for a query as:

$$\text{Precision}(q) = \frac{|P_{RELEVANT} \cap P_{RETRIEVED}|}{|P_{RETRIEVED}|}$$

where $P_{RELEVANT}$ denotes the set of pages that the user identified as the best matches for her query q, $P_{RETREVED}$ denotes the first twenty pages retrieved for query q and Precision (q) is the fraction of best q-matching pages in the first twenty results returned for q. This metric takes values from 0 to 1, with a value of 1 meaning that the top twenty pages returned for q are all best matches for q, and a value of 0 denoting that there is no best match for q between the top twenty retrieved pages. Thus, higher Precision values indicate better result quality.

Figure 8 aggregates the results by users, and demonstrates the overall effectives of our approach for every study participant. As we can see in the figure, the average precision that incorporates the user profile (e.g. PERSONALIZED) is much higher compared to the average precisions that rely on the queries' local and anchor text keywords respectively. Consequently, personalized query refinement yields significant improvement to the search quality over other query refinement techniques. In particular, we can see that our personalized query refinement technique outperforms the local and anchor text based refinement in all cases.

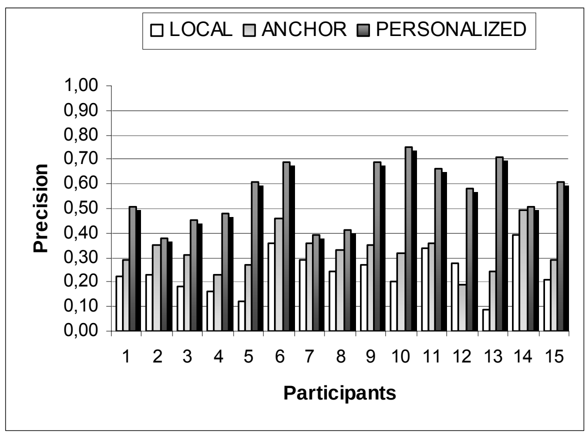

Figure 8. Precision @20 results for our test queries with respect to each of the refinement methods aggregated by users and averaged over each query in the dataset

A detailed look at the obtained results demonstrates that the degree of precision improvement depends upon the users' search patterns, i.e. the variety of topics that characterize the users' search profile. In particular, if we cross-examine the results reported in Figures 5, 6 and 8, we observe that the increased our model's accuracy in learning the user interests, the higher the probability that good alternatives will be detected for refining that user queries and thus, the better the quality of the results retrieved for those refined queries. This can be observed if we look at the results reported so far for user no.10. Note that according to Figure 5, user 10 indicated only two topical preferences in her examined search history. Based on the analysis of the user's search history, our model managed to learnt three topics for representing that user preferences and although the topic preference values that our model computed for these learnt topics were not analogous to the preference values of the user selected topics, nevertheless there was a strong semantic similarity between the actual and the learnt user preferences (cf. Fig 6). Therefore, given that our model managed to accurately capture that user's search interest it becomes evident that our personalized query refinement method that relies on the learnt user interests for selecting query alternatives, it managed to pick useful keywords for refining the user queries.

To further support our argument that as the learning accuracy of our user profiling method improves so does the precision of the obtained results for personalized refined queries, let's look at the case of user no. 14, who not only preferred a large number of topics (i.e. 8) in her recorded search trace but also attributed different interests to the same query across its multiple submissions (cf. Figure 7). According to Figures 5 and 6, the topics that our model estimated for representing that user's profile were marginally close to the topics that the user has selected for her considered queries. Therefore, since our personalized query refinement method relies on inaccurate topics for picking query alternatives it naturally occurs that the results retrieved for the user refined queries are of low quality. Based on the above, we may draw one conclusion; that the performance of our personalized query refinement technique strongly depends on the performance of our user profiling model.

Another interesting observation based on our findings is that our personalized query refinement technique is not dependent on the nature of the query detected topics. This can be attested in the improved retrieval performance of our method in all cases considered compared to the performance of the other query refinement methods. Like previously mentioned, user preferences were computed for specific ontology topics and they were not restricted to generic (i.e. top level) ontology topics. Therefore the nature of the topics computed for characterizing the user preferences does not influence our method's retrieval effectiveness. The same applies to the queries' nature, i.e. the performance of our method is not query-dependent since it relies on the query visited pages rather that on the query terms, for deriving the topical preferences of the user queries. This is verified in the improved retrieval performance for user no. 13 who although attributed semantically different topics to her search queries (cf. Figure 7), our model managed to accurately capture the user preferences based on the analysis of the past user searches (cf. Figure 6). Therefore, even though the user queries were vague and underspecified nevertheless the query visited pages helped us determined the latent user interests and therefore improve the user's queries in a meaningful manner. In contrast to the local and anchor refinement methods that explore query terms, our method which accounts for the topics in the query relevant pages yields improved retrieval performance.

The only factor upon which the performance of our query improvement technique generally depends is our model's efficiency in computing accurate user profiles. As we can see in Figure 8, our personalized query refinement model had the lowest retrieval improvement for users 2 and 14 for whom (according to Figures 5 and 6) our model had the lowest accuracy in learning their topical preferences. Therefore, we may conclude that as the accuracy of our user modeling method improves so does the retrieval performance. A less important observation in our results is that query refinement based on anchor text gives improved retrieval precision compared to the precision of the local query refinement technique. Nevertheless, our personalized query refinement method still greatly improves over the traditional ones overall. The average search quality improvement of PERSONALIZED query refinement, over ANCHOR and LOCAL query refinements, over all participants, is 74.5% and 136% respectively.

7 Related Work

Personalized search aims at the retrieval of information that is tailored to the user interests. Search personalization has attracted a substantial amount of work over the last few years, most of which address the challenge of user profiling. One approach to personalization is to have users explicitly describe their general search interests, which are stored as personal profiles [32], [47]. Many commercial systems rely on personal profiles to personalize search results by mapping Web pages to the same categories. For instance Google Personal [15] asks users to build their search profiles by selecting topics of preferences. This profile can then be used to personalize retrieved results by mapping query-matching pages to the same topics. Personal profiles, specified explicitly by the users have also been used to personalize the rankings of retrieved results [19], [2]. Our work is different from the above approaches in that our method does not require users to be directly involved in the profile generation process. Rather, our model aims at automatically capturing the user search preferences based on the analysis of their recorded search trace.

Nevertheless, there exist many works on the automatic learning of a user's preferences based on the analysis of her past clickthrough history [10], [34], [40] [35]. In [34] a user's preference is identified based on the five most frequent topics in the user's log data. Our work is different from this approach in that we consider all possible topics that describe a user's click history. Moreover, in [34] the authors limit their approach to the Web browsing paradigm and unlike our method they do not account for the semantic correlation between the pages and the issued queries. On the other hand, in [10], multiple TF-IDF vectors are generated, each representing the user's interests in one area. In [40] the authors employ collaborative filtering techniques for learning the user's preferences from both the pages the user visited and those visited by users with similar interests. While most of existing works share a common objective with our study (i.e. that of implicitly learning the user interests based on the analysis of their clickthrough data) our work is different from reported works in the following: we employ a topical ontology for building the user profiles as well as for automatically detecting the user interest that are latent behind their current queries for which there is no click data available. Additionally, we introduce a topic correlation value for evaluating our model's accuracy in learning user profiles, which considers the semantic similarity between the actual and the learnt user interests. Moreover, unlike previous works on search personalization that restrict their attempts to the ordering of search results, in this work we take a step further and address the issue of personalizing the query refinement process.

Mining the user's web logs in order to elicitate the user interests, has also been addressed in the context of e-commerce systems. In [53] the authors suggest a preferences mining model that relies on the learnt user interest in order to recommend users with products that meet their expectations and preferences. Building recommendations for items that may be of interest to the user has also been addressed in the work of [54] where the authors study how to assist users refine their preferences models. Although offering users with personalized e-commerce applications is a task that touches upon issues addressed in our study, nevertheless in this work we are mostly concerned with the web search paradigm where both user queries and online data are generally under-specified, semi-structured and of varying topics, intentions and preferences. However, we believe that our ontology-based model can be fruitfully employed in the context of e-commerce and e-services personalization.

Researchers have also studied ways of learning the user preferences at query time by means of techniques such as relevance feedback or query refinement. In [27] and [21] the authors examine various techniques for enabling the users specify how their queries should be expanded. Such techniques, offered through transparent interfaces rely on the use of relevance feedback, in which term suggestions are based on user relevance judgments of previously retrieved documents. While this approach has been shown to be effective in improving retrieval performance, it is difficult to implement in practice because of the reluctance of users to make the necessary relevance judgments [6].

A promising approach to personalizing search is to develop algorithms that infer intentions implicitly rather than requiring the users' intentions be explicitly specified. For an overview of such approaches, we refer the reader to the work of [24]. A multitude of implicit user activities have been proposed as sources of information for enhanced Web search, including the user's query [36] [37] and browsing history [40]. In [41] the authors explore the correlation between users, their queries and search results clicked, to model the user preferences. Likewise [42] employ rich models of user interests, built from both search-related information and information about the documents a user has read, created and/or emailed. In a recent study [39] we have also explored the automatic user modeling through the topical analysis of both the user queries and their obtained results. Our preliminary findings demonstrated that the semantic and topical analysis of the users' previous searches can convey accurate information about the user preferences. Based on these early findings, we improve our model and we concentrate our current study on how to personalize the query improvement process.

A lot of research in meta-search [33] [49] investigates mapping user queries to a set of categories or collections. However, the above techniques return the same results for a given query, regardless of who submitted the query. Our approach is different from the above in that we try to map user queries to particular topics based on the user's preferences for those topics. As such our query improvement technique retrieves results that are tailored to specific user profiles. To a further extend, we improve the user queries by suggesting terms that relate to both the user preferences and the query semantics.

Query personalization has attracted the interest of researchers in the past. Query personalization is the process of dynamically enhancing a query with related user preferences stored in a user profile with the aim of providing personalized answers. Most of the reported attempts in this area address the problem of query personalization in the context of databases [28], yet others pre-requisite that the documents in the users' search history are pre-classified to the desired categories against which the user queries are mapped [31]. The exploitation of such approaches in the context of Web searching is practically limited due to the dynamic nature of the flourishing Web data. Our approach differs from these studies in that we employ an ontology-based classification scheme for automatically mapping Web pages to their respective topics in the ontology. Moreover, we employ the topical ontology together with a topic-importance ranking function in order to estimate the user's degree of preference in particular topics. The most significant advantage that our approach demonstrates over existing techniques is that our refinement method accounts for non-stationary queries and tackles the changing user interests in an effective yet efficient manner.

Query expansion aims at selecting additional terms to be appended to the original query keywords in an attempt to improve retrieval performance. Accounting for the query semantics in an attempt to improve retrieval performance has attracted the interest of many researchers [7] [23] most of whom concentrated their efforts towards query sense disambiguation and thereafter query expansion [16]. Other techniques suggest query expansion based on either local [46] (i.e. results sets) or global [12] (i.e. thesauri) document analysis. Yet others proposed the utilization of lexical affinities to automatically refine queries [8], the usage of both the text surrounding the query terms in the search results as well as the text surrounding the query term in the document being read [50], while recently there have been efforts that suggest the utilization of conceptual ontologies [25] [9] for finding semantically related terms in order to improve retrieval efficiency.

Moreover, search engines have started offering short lists of search refinement suggestions in order to encourage the interactive narrowing of query search results. Such an example is the AltaVista Prisma tool[h]. In [3] the author experimentally evaluated how the searching behavior differed between users who employed AltaVista's Prisma assisted search tool and the users who operated without it. Obtained results indicate that the performance of the query reformulation process, *when applied*, was as effective as the average manual reformulation at locating relevant documents. Our work is different from the above approaches in that we are primarily interested in generating search suggestions that are tailored to the user's specific profiles, i.e. preferences. Moreover, we opt for the automatic refinement of search queries, i.e. without the user involvement. Therefore, in our work we are essentially concerned with the automatic identification of the user preferences and their encapsulation in the generated refined queries. To the best of our knowledge, the only reported study that brings together search personalization and automatic query expansion is the work of Chirita et al. [11] in which query refinement relies on desktop data for extracting keyword expansion terms. Desktop data is what the authors call the Personal Information Repository (PIR) and represents a rich source of profiling information. By expanding queries with terms that are extracted from the user's PIR, the search output is implicitly personalized. Although our work shares a common motivation with the work in [11], nevertheless our implementation is totally different, since we rely on the users' recorded search behavior rather than on their personal collection of data. Moreover, our approach is different in that user topic preferences are determined based on the content of real web pages and as such they represent the particular interests of the user while interacting with the web, which may be different to the general user interests communicated via the data she stores in her workstation.

With respect to expanded query visualization, the vast majority of existing works provide alternative query formulations as a list of keywords returned to the user together with the search results. In [43] the authors suggested the graphical representation of queries as a means to help users find what they are looking for. In a recent work [29] we studied the automatic organization of refined queries into graphical forms and we experimentally evaluated the users' perception of the generated query graphs' informativeness. Obtained results demonstrated that structured queries have a significant potential in assisting users clarify their vague information needs. In this article, we build upon previous work and we explore the automatic structuring of queries from a personalization perspective.

In summary, we believe that our work on personalized query refinement touches upon issues addressed in previous studies and it further expands them by suggesting novel ways of improving the user experience when querying the Web. As such as deem our work to be complementary to existing techniques. The contribution of our work lies in the exploitation of a topical ontology for the automatic learning of the user preferences and the subsequent incorporation of the learnt preferences in the users' future searches. The use of ontologies for improving personalized searches would allow search engines adapt to changing conditions, changing user search preferences and different search settings.

---

[h] http://newsbreaks.infotoday.com/nbreader.asp?ArticleID=17139

8 Concluding Remarks

In this article, we have investigated the personalized query refinement problem and we presented a novel framework, which uses a topical ontology for improving the user issued queries with words that are relevant to both the user preferences and the query semantics. In particular, we first proposed an ontology-based model for automatically learning the user topic preferences based on the analysis of the user's observable clickthrough data. Our model explores the users' search patterns and employs a topic importance ranking function for measuring the degree of the user's preferences in the identified topics. We experimentally evaluated the learning accuracy of our user profiling method and we showed that our model manages to capture the user interests quite accurately, when these span in a relatively few number of topics (i.e. up to 6 topics). We then described how the learnt user preferences can be fruitfully explored in the query refinement process, in an attempt to provide web information seekers with useful terms for reformulating their search requests. In particular, we described how our method uses the learnt user preferences for detecting the topical intention that is hidden behind a search query for which there might not always be clickthrough data available. We then described how to use the ontology for identifying alternative query terms that match both the query semantics and the user preferences. Finally, we introduced a novel representation scheme for visualizing the refined queries in order to assist the user make informative decisions about whether to issue a suggested query or not. To evaluate the effectiveness of our personalized query refinement technique in improving retrieval performance, we carried out an experimental study. In our evaluation, we compared the effectiveness of our technique to the effectiveness of other query improvement methods in delivering qualitative search results. Our comparative evaluation demonstrates that our technique significantly outperforms existing search improvement methods in retrieving results that are of good quality and highly relevant to the user intentions.

We now discuss some advantages that our approach exhibits compared to other search improvement techniques. Firstly, given that our query refinement technique explores the user preferences, in addition to the query semantics, we believe that our method can work well in the highly changing environment of the real search engines, as it allows a level of personalized refinement even when the user interests are volatile, vague or they project several topics. Another advantage of our approach is that it uses a built-in topical ontology to automatically identify the user preferences, the query intentions and the web pages conceptual content. As such, it not only guarantees consistency between the user profile and the web data representations, but it also ensures that search improvement can be achieved without any user intervention besides that of query issuing. However, our method is not tightly intergraded with the particular ontology but rather it can be successfully deployed for any ontology one would like to use. In overall, we believe that an important contribution of our work to the web engineering community is the exploitation of topical ontologies for achieving improved web searches. In particular, our work demonstrates the impact that topical ontologies can have on different aspects of web searching, such as search personalization, query improvement, offering topic-specific rankings, facilitating the automatic categorization of web pages, the semantic processing of web data and the extraction of thematic keywords from the pages' contents. We therefore believe that existing ontologies can be fruitfully explored by the web community for improving web services and thus for assisting web users experience successful web interactions.

With respect to some practical considerations about the feasibility of integrating our ontology-based model in existing web applications and services, we should refer to the following. Building an ontology is at first sight a burdensome and time-consuming task. Nevertheless once built, the ontology can serve multiple purposes and thus it compensates for the effort put towards its implementation. In practice, it took us nearly three months to integrate the SUMO and MWND ontologies with Dmoz top-level topics, but having built the ontology we are able to automatically carry out several tasks such as content-based web data classification, topical-based ordering of search results, modeling user search preferences, deciphering web query intentions, refining user requests with semantically similar and user-preferred terms and so on. Considering that without utilizing a single ontology, we would not be able to carry out all of the above tasks in such an effective manner and also that for every task we would need to develop different techniques, we believe that the time spent for building the ontology was not so significant compared to the ontology's contribution in all the above tasks. Still, in the absence of or the reluctance to use an ontology, our method would work well with a simple lexical hierarchy instead, such as WordNet, which is freely available, comes in different forms and provides many tools for its exploitation. Another option, besides an ontology or WordNet would be to turn to a web directory or a faceted classification scheme in order to identify the thematic contents of the web data.

Besides the practical considerations pertaining to the availability of an ontology, our method could be readily deployed for a number of web applications that operate upon classified web data. One such possibility would be to integrate our user profiling module in e-commerce recommendation systems. In this scenario, our user modeling approach could be integrated in e-commerce servers where data about users is stored in order to analyze the data recorded about the users' transactions with the system and learn the preferred user products and services in a much similar way as the user search preferences are learnt in the present work. Then, based on these learnt preferences a recommendation system could suggest users with useful products or rather it could customize the e-commerce portals presentation and structure in a personalized manner so as to meet specific user or user group interests. In this respect, collaborative filtering methods may be integrated with our profiling mechanism in order to increase the accuracy of the derived recommendations. Moreover, in the process of building customer profiles a specialized ontology of product categories, names and features would be useful and the latter could be built based on a similar approach we followed in the present work. Another possibility would be to integrate our query refinement method together with the topical ontology in a focused-crawling module in order to provide the latter with alternative queries to be issued in online databases of hidden web content.

There are a number of promising directions for future investigation. In particular, we are considering several parameters that could be exploited for extending our user profiling method with rich models that capture the complete user activity while visiting a page; such as email, print and/or save of a page, focus points of the page, etc. Also we suspect that a query refinement model that mines the anchor text of the query matching pages and encapsulates the user preferences in the mining process, it can come up with query alternatives that are tailored to the user interests and which can achieve personalized qualitative search results. Another interesting issue for future investigation is the deployment of our personalized query refinement technique into a real-life, large-scale web search engine with millions of users and queries. In such an attempt, the challenges that arise concern the interface design, the user interaction with the system, the dynamic ontology updates, the re-ranking of the refined query results, the discrimination between long-term and short-term user preferences, and many others. We hope that our article stimulates interest to develop interesting applications for personalized query refinement, and that web search engines start investigate some of the ideas to further improve their overall search performance.

References

1. Agichtein E., Brill E., Dumais S. and Rango R. Learning user interaction for predicting web search result preferences. Proc. of SIGIR Conf., 2006.
2. Aktas M. Nacar N. and Menczer F. Personalizing PageRank based on domain profiles. Proc. of KDD Workshop on Web Mining and Web Usage, pp83-90, 2004.
3. Anick P. Using terminological feedback for web search refinement: a log-based study. Proc. of World Wide Web Conf., pp89-95, 2004.
4. Barzilay R. Lexical chains for text summarization. Master's Thesis, Ben-Gurion Univ, 1997.
5. Beitzel S. Understanding and Classifying Web Queries. Ph.D. Dissertation, IIT, 2006.
6. Belkin N.J., Cool C., Head J., Jeng J., Kelly D., Lin S., Lobash L., Park S.Y., Savage-Knepshield P. and Sikora C. Relevance Feedback versus Local Context Analysis as Term Suggestion Devices: Rutgers' TREC-8 Interactive Track Experience. TREC 2000.
7. Billerbeck B., Scholer F., Williams H.E. and Zobel L. Query expansion using associated queries. Proc. of ACM CIKM Conf., 2003.
8. Carmel D., Farchi E., Petruschka Y. and Soffer A. Automatic query refinement using lexical affinities with maximal information gain. Proc. of 25th ACM SIGIR Conf., pp283-290, 2002.
9. Celik D. and Elci A. Discovering and scoring of semantic web services based on client requirement(s) through a semantic search agent. Proc. of 30th IEEE Computer Software and Applications Conf., pp273-278, 2006.
10. Chen L. and Sycara K. Webmate: a personal agent for browsing and searching. Proc. of 2nd Intl. Conf. on Autonomous Agents & Multiagent Systems, pp132-139, 2004.
11. Chirita P.A., Firan C.S. and Nejdl W. Personalized Query Expansion for Web. Proc. of SIGIR Conf., 2007.
12. Frakes W.B. and Baeza-Yates R. Information Retrieval: Data Structures and Algorithms. Prentice Hall, New Jersey, 1992.
13. Gauch S., Chafee J. and Pretschner A. Ontology-based personalized search and browsing. Web Intelligence and Agent Systems, Vol.1, No.3-4, pp.219-234, 2004.
14. Gliozzo A., Strapparava C. and Dagan I. Unsupervised and Supervised Exploitation of Semantic Domains in Lexical Disambiguation. Computer Speech and Language, 18(3):275-299, 2004.
15. Google Personal: http://labs.google.com/personalized
16. Gong Z., Cheang C.W. and Hou C. Web query expansion by WordNet. Proc. of DEXA Conf., 2005.

17. Gulli A. and Signorini A. The indexable Web is more than 11.5 billion pages. Proc. of World Wide Web Conf., 2005.

18. Jansen B.J., Spink A. and Saracevic T. Real life, real users, and real needs: A study and analysis of user queries on the Web. Information Processing & Management, 36(2):207-227, 2000.

19. Jeh G., Widom J. Scaling personalized web search. Proc. of 12th Intl. WWW, pp271-279, 2003.

20. Joachims T., Granka L. Pang B., Hembrooke H. and Gay G. Accurately Interpreting Clickthrough Data as Imlicit Feedback. Proc. of ACM Conf. on Research and Development in Information Retrieval (SIGIR), 2005.

21. Harman D. Towards Interactive Query Expansion. Proc. of SIGIR Conf., pp321-331, 1988.

22. Haveliwala T. Topic-sensitive PageRank. Proc. of World Wide Web Conf., 2002.

23. Hawking D. and Craswell N. Overview of the TREC-01 Web Track. 10th Retrieval Conf., 2001.

24. Kelly D. and Teevan J. Implicit feedback for inferring user preference: a bibliography. SIGIR Forum 32(2), pp18-28, 2003.

25. Khan L., McLeod D. and Hovy E. Retrieval effectiveness of an ontology-based model for information selection. VLDB Journal, Vol.13, pp71-85, 2004.

26. Kraft R. and Jien J. Mining anchor text for query refinement. Proc. of World Wide Web Conf., pp666-674, 2005.

27. Koenamann J. and Belkin N. A case for interaction: A study of interactive information retrieval behavior and effectiveness. Proc. of CHI Conf., pp205-212, 1996.

28. Koutrika G. and Ioannidis Y. Personalized queries under a generalized preference model. Proc. of the ICDE Conf., 2005.

29. Kozanidis L., Tzekou P., Zotos N. Stamou S. and Christodoulakis D. Ontology-based adaptive query refinement. Proc. of 3rd WebIST Conf., 2007.

30. Krikos V., Stamou S., Kokosis P., Ntoulas A., and Christodoulakis D. DirectoryRank: Ordering Pages in Web Directories. Proc. of 7th ACM Intl. Workshop on Web Information and Data Management, pp17-22, 2005.

31. Liu F., Yu C., and Meng W. Personalized web search by mapping user queries to categories. Proc. of CIKM Conference, pp558-565, 2002.

32. Pazzani M., Muramatsu J., Billsus D. Syskill & Webert: Identifying Interesting Web Sites. Proc. of 13th National Conf. on Artificial Intelligence, Portland, pp54-61, 1996.

33. Powel A.L., French J.C., Callan J.P. and Connell M. The impact of database selection on distributed searching. Proc. of SIGIR Conf., 2000.

34. Pretschner A., Gauch S. Ontology-based personalized search. Proc. of 11th IEEE Intl. Conf. on Tools with Artificial Intelligence, pp391-398, 1999.

35. Qui F. Cho J. Automatic Identification of User Interest for Personalized Search. Proc. of 15th Intl. World Wide Web Conf., pp727-236, 2006.

36. Shen X. and Zhai C.X. Exploiting query history for document ranking in interactive information retrieval. Proc. of SIGIR Conf., pp377-378, 2003.
37. Speretta M. and Gauch S. Personalizing search based in user search history. CIKM Conf., 2004.
38. Stamou S., Ntoulas A., Christodoulakis. TODE: An Ontology based model for the dynamic population of Web directories. Data Management with Ontologies: Implementations, Findings and Frameworks, 2007.
39. Stamou S. and Ntoulas A. Search personalization through query and page topical analysis. Journal of User Modeling and User-Adapted Interaction, Vol.19. nos. 1-2, 2009
40. Sugiyama K., Hatano K., Yoshikawa M. Adaptive web search based on user profile constructed without any effort from users. Proc. of 13th Intl. World Wide Web Conf., pp675-684, 2004.
41. Sun J., Zeng H., Liu H., Lu Y., Chen Z. CubeSVD: A novel approach to personalized web search. Proc. of 14th Intl. World Wide Web Conf., pp382-390, 2005.
42. Teevan J., Dumais S., Horvitz E. Personalizing search via automated analysis of interests and activities. Proc. of 28th Intl. Conf. on Research and Development in Information Retrieval, pp449-456, 2005.
43. Tomita J. and Kikui G. Interactive web search by graphical query refinement. Proc. of World Wide Web Conf., 2001.
44. Jiang J., Conrath D. Semantic Similarity based on Corpus Statistics and Lexical Taxonomy. Proc. of Intl. Conf. on Research in Computational Linguistics, 1997.
45. Wu X. and Palmer M. Web semantics and lexical selection. In 32nd ACL Meeting, 1994.
46. Xu J. and Croft B. Query expansion using local and global document analysis. Proc. of SIGIR Conf., pp4-11, 1996.
47. Yahoo! Inc. *MyYahoo* http://my.yahoo.com.
48. Yang Y. and Liu X. Re-examination of text categorization methods. Proc. of SIGIR Conf., 1999.
49. Yu C., Meng W., Wu W. and Liu K. Efficient and effective metasearch for text databases incorporating linkages among documents. Proc. of ACM SIGMOD Conf., 2001.
50. Kawashige T., Oyama S., Ohsima H. and Tanaka K. Context Matcher: Improved web search using query term context in source document and in search results. Proc. of APWeb., pp486-497, 2006.
51. Song Y.I., Han K.S. and Rim H.C. A term weighting method based on lexical chain for automatic summarization. Proc. of 5th CICLing Conf., pp636-639, 2004.
52. Broder A.Z., Glassman S.C., Manasse M. and Zweig G. Syntactic clustering of the web. Proc. of 6th Intl World Wide Web Conf., pp1157-1166, 1997.
53. Holland S., Ester M., Kiessling W. Preference Mining: A Novel Approach on Mining User Preferences for Personalized Applications. Proc. of PKDD Conf., pp204-216, 2003.
54. Chen L., Pu P. Preference-Based Organization Interfaces: Aiding User Critiques in Recommender Systems. Proc. of User Modeling, pp77-86, 2007.