Evaluation of Ontology-based User Interests Modeling

Zhongming Ma  Olivia R. Liu Sheng  Gautam Pant
David Eccles School of Business
The University of Utah
zhongming.ma@business.utah.edu  olivia.sheng@business.utah.edu  gautam.pant@business.utah.edu

Abstract

Deriving users’ interests from their online searching and browsing behaviors is an important research direction with several applications in content search and management. Manually edited Web directories, such as Open Directory Project1 (ODP) or Yahoo!2 directory, provide ontology of concepts (categories) along with pages relevant to those categories. Aiming to evaluate and compare the performance of different cues in searching and browsing activities for user interests modeling, we used four inputs (search query, expanded query, snippet, and page content) and mapped each of them into a set of ODP categories. Then we automatically identified an output category as representative of user interest that initiated a search. Through a controlled experiment, we compared the performance of different inputs in interest mapping using two metrics - hit rate and average hit index. We found that the use of page content achieved the best results, i.e. highest hit rate and lowest average hit index. Also an expanded query (the original query with a few additional terms) was a better input to identify user interests than the original query and the snippet. The expanded query produced hit rates 21-121% higher than those achieved by the original query and had 34% lower average hit index than that by the original query.

Keywords: user interests, query expansion, Web search, Open Directory

1. Introduction

The Web is an extremely large, diverse, and dynamic source of information. To deal with the information overload, people use search engines to locate Web pages that contain desired information. Besides searching, another fundamental Web activity for finding online information is browsing. Behavior-based user profiling models a user’s interests based on his/her activities [11]. The types of information previously used for user profiling include inputs to a text editor, online navigation, usage of a command-based interface [9], and online searching. User’s interactions with a Web browser during searching and browsing are an obvious source of such profiling. A user profile that represents a user’s interests can serve many purposes, such as feeding a personalization or recommendation system [12, 3].

To obtain insights into how to improve and apply ontology-based user profile modeling methods, it is useful to know how different cues affect the performance, e.g. accuracy, of profile modeling. The goal of this paper is to evaluate and compare the affect of four inputs (search query, expanded query, snippet, and page content) in identifying user interests. In particular, we present an automatic query expansion approach and examine how the expanded query performs better than the original query in modeling interests. We map each input into a group of categories from ODP ontology and compare the performance of different inputs along two metrics: hit rate and average hit index. Except for the search query (a direct input from a user), other three inputs are implicitly derived. The original query is augmented with a few semantically related terms through a query expansion technique to create an expanded query. The snippet is a small amount of text that often accompanies each search engine result (see Fig. 4). Finally, the page content, corresponding to search engine results that have been clicked and viewed by a user, is also used as an input.

1 http://www.dmoz.com
2 http://www.yahoo.com
Through our experiments with 14 subjects, we found that using page content achieved the highest hit rates and lowest average hit index; the expanded query outperformed the original query and snippet on the two metrics. Compared with the original query, the expanded query produced 21-121% higher hit rates and 34% lower average hit index.

2. Related Work
A straightforward approach to identifying a user’s interests in Web content is to obtain explicit ratings on Web pages from the user (e.g., [12] [18].) However, expecting a user to provide explicit feedback is not desirable since it requires additional user effort and keeps the user from performing his/her real work [10]. According to observations by Carroll and Rosson [1], users are reluctant to provide explicit feedback despite their recognition of its long-term benefits.

Therefore a number of studies use implicit feedback for inferring user interests. For a Web user, implicit feedback includes page clicks, time spent on a page, the number of mouse clicks, and the amount of scrolling on a page. Sugiyama et al. [17] and Gauch et al. [3] build and use a user profile based on browsing history (i.e., clicked pages) to re-rank results from a search engine.

**Ontology** is a formal specification of concepts and relationships amongst them [4]. Using ontology to represent user interests allows communication and knowledge sharing among people. The Open Directory Project (ODP) ontology is a directory of 5 million URLs manually categorized into 590,000 categories. Fig. 1 illustrates a portion of ODP ontology, where **Computers** is a depth-one category and **C++** and **Java** are categories at depth four. We call **Computers/Programming/Languages** the parent category of category **C++** or **Java**, and **Computers/Programming/Languages/C++** is a child category of **Computers/Programming/Languages**.

Categories of ontology have been used to represent user interests in the past. With periodically collected visit history (clicked Web pages) Gauch et al. [3] and Trajkova and Gauch [19] generate a set of ontology categories by mapping the content of each page to a group of categories in Magellan ontology. After mapping all of the visited pages collected over a certain time period into a user profile, the profile reflects an overall user interests in the form of ontology categories. Liu et al. [8] build a user profile that consists of previous search query terms and five words surrounding a query term in each clicked Web page after the query is issued. With the profile, they map a user’s search query to several depth-two ODP categories.

Query expansion is the process of augmenting a query provided by a user with other words or phrases in an effort to improve search effectiveness. Query expansion was originally applied in information retrieval (IR) to solve the problem of word mismatch that arises from differences in the words used by search engine users to refer to a concept and the words used by content authors to describe the same concept [20]. IntelliZap system [2] generates additional query terms based on the text surrounding the original query terms in a document the user is reading.

Related work suggests the use of various cues could be effective for ontology-based user profile modeling. The four inputs in this study are all easy to retrieve. Compared with the other three inputs, page content contains many more terms which may be helpful in identifying user interests. However, the other three inputs can be obtained even before a user views a page. Evaluation of such cues based on their effectiveness for ontology-based user interests modeling is important but lacking in past studies.
3. Research Design

Fig. 2 depicts an overview of our evaluation framework. A set of search tasks related to the computer domain is predefined by two domain experts. For each task, a user issues a query and a search system that we developed retrieves search results from Google\(^3\). Then our query expansion approach takes the initial query and up to ten search results to automatically derive additional query terms. Once the user clicks a result and views the corresponding page content, the two other inputs – snippet and page content – can be identified. Then we can compute a cosine similarity between an input, such as a query, and representations of ODP categories to produce a set of mapped ODP categories.

Since in this study we focus on comparing the performance of four inputs for interests mapping, we have predefined search tasks and thus a domain expert can identify a set of relevant categories in ODP that are representatives of the interests that initiate each search task. This procedure is different from prior studies [3, 8, 11] that learn user profiles over a period of time and may need subjects to verify the relevance of ontology categories in their profiles. For example, to make sure the profile represents user interests, some researchers ask subjects to manually verify the profiles after many queries have been issued [3, 8]. Different from the previous studies, in our experiment the use of ODP (or any ontology) is transparent to the subjects. In the current experiment the evaluation program automatically verifies whether an output category represents an interest according to the relevant categories (identified by a domain expert) and computes performance metrics. Table 1 presents some sample search tasks and corresponding relevant categories. Next we describe how ODP category profiles are created as representations of these categories followed by a description of query expansion technique.

<table>
<thead>
<tr>
<th>Search task</th>
<th>Relevant category</th>
</tr>
</thead>
<tbody>
<tr>
<td>Find out dates and city for Twentieth National Conference on Artificial Intelligence.</td>
<td>computers/artificial intelligence/conferences and events</td>
</tr>
<tr>
<td>Find a Web page from where you can download JDK 5.0</td>
<td>computers/programming/languages/java</td>
</tr>
<tr>
<td>Find a Web page introducing C++ multiple inheritance with specific example(s).</td>
<td>computers/programming/languages/c++</td>
</tr>
<tr>
<td>Find the home page of Tom Mitchell who is a professor of Artificial Intelligence and Machine Learning.</td>
<td>computers/artificial intelligence/computers/artificial intelligence/learning computers/artificial intelligence/people</td>
</tr>
</tbody>
</table>

Table 1 Sample search tasks and matched categories

3.1 Generating category profile

ODP is a categorical collection of millions of manually edited URLs. Each ODP category contains URLs that point to external Web pages that have been considered relevant to the category by human editors. These URLs are accompanied with manually composed titles and summaries that, we believe, can accurately represent the corresponding Web page contents. Shen et al. [16] report that classification using manually composed summarization in LookSmart Web directory achieves higher accuracy than using the content of Web pages.

Fig. 3 shows an ODP category, Computers/Artificial Intelligence, and some URLs (manually) categorized under it. We build a category profile for an ODP category by concatenating up to 30 pairs of titles and

\(^3\) http://www.google.com
summaries of URLs listed under the category. For a category at depth \( n \) (\( n=2 \) or 3), such as Computers/Artificial Intelligence, the category profile contains not only titles and summaries of URL links directly under it, but also those under its child categories at depth \( n+1 \), such as Computers/Artificial Intelligence/Machine Learning and Computers/Artificial Intelligence/Natural Language.

We represent the category profile as a term vector with term frequencies as weights as shown in Fig. 3. Thus the category profile includes category profiles of its child categories.

The range of categories that may be used to model user interests include all categories up to depth four that appear under 14 depth-one (excluding Regional and World) categories in the ODP. We remove a category if the category profile contains less than 20 terms, giving us a total of 22,079 categories which is much greater than those used in prior studies [3, 8, 19]. We generate category profiles for all these 22,079 categories by removing stops words, applying the Porter stemming algorithm [13] and removing terms with frequency of one. Finally, the profile is used for computing a cosine similarity with an input, e.g. a query or snippet.

### 3.2 Query expansion approach

Besides the word mismatch problem previously mentioned, Web users tend to submit short queries with an average length of about two [6, 7]. Hence, it may be worthwhile to augment the query with a few semantically similar or related terms so that the output categories generated by the expanded query are more likely to cover the user interests. According to Distribution Structure theory [5], in a language terms do not occur arbitrarily relative to each other; a term occurs at a certain position relative to certain other terms. Based on the concept of co-occurrence, Riloff and Shepherd [14] present a bootstrapping algorithm that starts with a few given seed words belonging to a specific domain and find out a larger number of domain-specific semantically related lexicons from a corpus.

Co-occurrence has been applied to query expansion of Web search in prior studies [e.g. 2, 8]. In this paper we identify co-occurring nouns from search results (titles and snippets) to derive additional terms. Our automatic query expansion approach does not require any additional input from a user. When a query is issued, one of our programs finds the \( N \) most frequently co-occurring nouns from up to ten pairs of page titles and snippets on the first result page. An advantage of this method is that we can find semantically similar or related terms to the original query without asking the user for any input, such as a feedback or a corpus. Being able to find semantically similar or related terms is especially helpful when the original query is or contains a proper noun, such as “NLP” and “Tom Mitchell”. Then we compute the similarity between the expanded query\(^5\) and each of the category profiles.

Table 2 shows some sample queries issued by subjects in our experiment and the corresponding (three) frequently co-occurring nouns. The query “IJCAI”, an abbreviation of “International Joint Conference on Artificial Intelligence,” did not generate any output category. However, with the expanded query, the output categories contained a relevant category. Similarly, the expanded query terms (machine, magazine, and intelligence) provided more contextual information for discovering user interests when searching for “Tom Mitchell AI.”

---

\(^4\) We do not generate category profiles for depth-one categories because they are too general to represent an interest.

\(^5\) An expanded query is a combination the original query and the \( N \) most frequently co-occurring nouns.
4. Experiments

Using Google APIs, we developed a system that sent user queries to Google and displayed the results retrieved from the search engine. The system also recorded queries issued and URLs clicked by a subject and logged the time when such an event occurred. Fig. 4 is a sample output for query “Twentieth National Conference on Artificial Intelligence.”

We recruited 14 subjects. All of them were graduate or senior undergraduate students in the computer science department of our university or professionals in software industry. Two of our authors who had industrial experience related to computer science acted as domain experts to predefine a set of search tasks. All search tasks are related with the computer domain and each task specifies the nature of information and the number of Web pages that need to be found. Our analysis in this paper is based on nine tasks that allowed users to type in their own queries with any number of terms. During the experiment, each of the search tasks was shown in a search task area located at top of the GUI. A user could proceed to the next task by clicking the “Next” button.

We considered a page to be of interest to a user if the viewing time was longer than a threshold (we chose three seconds), and one of our programs automatically retrieved the content up to 10KB for each page. If a user viewed multiple pages after a query, the program concatenated the content of these pages whose viewing time was greater than the threshold to form one master document which was used to generate a set of output ODP categories. Based on the relevant categories defined by a domain expert for each task, the evaluation program automatically determined which category in the output was relevant and recorded the index of the relevant category. We considered a relevant category in a set of output categories to be a hit (a term has been used in recommender system literature, e.g. [15]) and we called the index of the relevant category the hit index. We define the hit rate and the average hit index for a given input as:

\[
\text{hit rate} = \frac{\sum_{i=1}^{N_i} \sum_{j=1}^{N_j} h(i, j)}{I}, \quad \text{average hit index} = \frac{\sum_{i=1}^{N_i} \sum_{j=1}^{N_j} h(i, j)}{I}
\]

If there were multiple relevant categories for a task, our evaluation program recorded the index of the first relevant category.

A smaller index was better because the relevant category was ranked higher in the output.
where \( h(i, j) \) is 1 if the set of output categories contains a hit, and 0 otherwise; \( hi(i, j) \) is the hit index for a subject (i) and a given task (j). If the output did not contain a relevant category (no hit), we set the hit index to be the number of output + 1. \( N_s \) is the number of subjects and \( N_t \) is number of search tasks. Since we had 14 subjects and provided 9 tasks to each of them, the total number of performed tasks or the total number of times interests identified, \( I \), was 122, given three subjects gave up on four of the tasks.

5. Result Analysis

When expanding a query, we tested different number of expanded terms ranging from one to four. We found that with three expanded terms we achieved both highest hit rate and best (lowest) average hit index. Thus we chose query expansion with three additional terms in our experiment.

When producing a set of ODP categories for each input, we changed the number of output categories from 1 to 20 as shown on X-axis in Fig. 5. We found that, as compared to using the original query, the expanded query achieved a higher hit rate (21% to 121% higher depending on the number of output categories). Similarly, the page content obtained a (16% to 69%) higher hit rate than the snippet. In addition, the expanded query outperformed the snippet on hit rate.

Table 3 lists the average hit index and ±1 standard error for each input. The results are based on the number of output categories at 20. This table shows that the page content had the best average hit index, and the expanded query was the second best input, better than the original query and snippet. A snippet retrieved from Google consisted of the original query and several words surrounding the query terms in a page. As we see from Fig. 4, a snippet is longer than the expanded query. Thus we concluded that the snippet contained more noise than the expanded query for the purpose of identifying user interest(s). Another reason that the expanded query outperformed the original query and snippet was that the expanded terms were semantically similar or related to the original query terms. A reason for the original query to have a larger average hit index was that the set of output categories generated by original query was less likely to contain the relevant categories than using other inputs; this situation is also suggested by a lower hit rate for the original query as seen in Fig. 5.

<table>
<thead>
<tr>
<th>Input</th>
<th>Original Query</th>
<th>Expanded Query</th>
<th>Snippet</th>
<th>Page Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hit index</td>
<td>11.06 ± 0.79</td>
<td>7.34 ± 0.64</td>
<td>9.28 ± 0.71</td>
<td>5.56 ± 0.58</td>
</tr>
</tbody>
</table>

Table 3 Comparison of average hit index for different inputs

6. Conclusion

A user’s online activities, such as searching and browsing, are important sources to derive his/her interests with respect to information needs. The automatic identification of user interests can result in a user profile which may be applied for content search and management. In this study, we identified user interests by mapping the artifacts of online search behaviors onto the ODP ontology. We focused on evaluating and comparing four different inputs (search query, expanded query, snippet, and page content) in interests mapping using the hit rate and average hit index metrics. Through our experiment with 14 subjects we found that using the page content achieved the best performance in terms of hit rate and average hit index. The expanded query proved to be the second best input, better than the original query and snippet. The results indicate that (1) the expanded terms obtained by our query expansion approach
are semantically similar or related to the original query terms, as a result improving the hit rate and average hit index; (2) a snippet contains more noise than the expanded query, leading to lower hit rate and larger average hit index.

In this paper, we have teased out and compared the affects of various inputs in modeling user interests based on ODP ontology. All of the considered inputs can be easily extracted from a Web log at either the client side or the server side. A comparative study of such inputs in modeling user interests is timely given the widespread use of search engines and the availability of large-scale ontology such as the ODP. In this study we limited user interests in computer domain. As future research we would like to compare the cues in different domains, develop and evaluate applications that are based on user interest models derived from users’ real searches instead of pre-defined tasks, and examine how a combination of different cues can improve the performance.

7. References