Peter Brusilovsky, Charles Callaway, Andreas Nürnberger (Eds.)

PIA 2005

Workshop on New Technologies for Personalized Information Access

Proceedings of the 1st International Workshop

Held in conjunction with:

10th International Conference on User Modeling (UM'05)
Edinburgh, UK, July 24th to 30th, 2005

These proceedings are available online at the workshop web site:

http://irgroup.cs.uni-magdeburg.de/pia2005/

Preface

Information access is one of the hottest topics in creating the future information society and it has become even more important since the advent of the Web. On one hand, our society relies more and more on information, both for professional and personal goals. Information is nowadays considered as one of the most valuable and strategic goods: knowing the right information, at the right moment, as soon as it is available is a "must" for all of us. On the other hand, the amount of available information, especially on the Web and in modern Digital Libraries, is increasing tremendously over time.

In this context, the importance and role of user modeling and personalized information access are increasing. Equipped with user modeling tools capable of comprehending specific user information needs, new retrieval tools will be able to effectively filter out irrelevant information, to rank information in the most suitable way, to compare the contents of different documents, to personalize information presentation, and to adequately tailor man-machine interaction.

The new challenges have motivated a range of new technologies for personalized information access within all information access paradigms – from classic "ad-hoc" information retrieval to information filtering, browsing, presentation, and visualization. New creative ideas have emerged in a number of old and new research communities including user modeling, machine learning, adaptive hypermedia, digital libraries, the Semantic Web, human-computer interaction, and information visualization.

The goals of the workshop are to intensify the exchange of innovative ideas between the different research communities involved, to provide an overview of current activities in the area of personalized information access, and to point out connections between them. The workshop focuses especially on researchers that are working on ontologies, computational linguistics, user modeling and profiling, user adaptive interfaces, digital libraries, and their combination.

17 papers from 10 countries were submitted to the workshop. After careful reviewing, the program committee selected 5 full and 4 short papers for presentation at the workshop. These papers are included in the proceedings in alphabetic order. The selected papers discuss a range of issues related to personalized information access. We hope that this collection of papers will be useful for researchers and practitioners in this emerging area.

Peter Brusilovsky, Pittsburgh, USA
Charles Callaway, Edinburgh, UK
Andreas Nürnberger, Magdeburg, Germany

Organization

Organizers

Peter Brusilovsky – University of Pittsburgh, USA
Charles Callaway – University of Edinburgh, UK
Yannis Ioannidis – University Of Athens, Greece
Andreas Nürnberger – University of Magdeburg, Germany

Program Commitee

Stavros Christodoulakis – Technical University of Crete, Greece
Cristina Conati – University of British Columbia, Canada
Owen Conlan – Trinity College, Dublin, Ireland
Ernesto William De Luca – University of Magdeburg, Germany
Marcin Detyniecki – Laboratoire d'Informatique de Paris 6 (LIP6), France
Susan Dumais – Microsoft Research, Redmond, USA
Aldo Gangemi – Laboratory for Applied Ontology (ISTC–CNR), Rome, Italy
Susan Gauch – University of Kansas, USA
Andreas Hotho – University of Kassel, Germany
Antonio Krüger – University of Münster, Germany
Anton Leuski – University of Southern California, USA
Bernardo Magnini – ITC/IRST, Trento, Italy
Kathleen F. McCoy – University of Delaware, USA
Claudia Niederée – Fraunhofer IPSI, Germany
Christos Papatheodorou – Ionian University, Greece
Giovanni Semeraro – University of Bari, Italy
Alan Smeaton – Dublin City University, Ireland
Barry Smyth – University College Dublin, Ireland
Amy Soller – Institute for Defense Analyses, USA
Carlo Strapparava – ITC/IRST, Trento, Italy
Oliver Streiter – University of Kaohsiung, Taiwan
Andreas Witt – University Bielefeld, Germany

Sponsoring Institutions

DELOS Network of Excellence on Digital Libraries

Table of Contents

Investigating Users' Needs and Behavior for Social Search

Jae-wook Ahn[1], Peter Brusilovsky[1], and Rosta Farzan[2],

[1]School of Information Sciences and [2]Intelligent Systems Program
University of Pittsburgh, Pittsburgh PA 15260, USA
jahn@mail.sis.pitt.edu, peterb@pitt.edu, rosta@cs.pitt.edu

Abstract. Traditional Web search systems have limitations due to its unrealistic assumption on users' query formulation and lack of context-sensitivity. To overcome these limitations, we designed and implemented a social search system which is based on a social adaptive navigation system Knowledge Sea and exploits the past usage history of users. By conducting a survey and transaction log analysis, we could observe users' strong attitudes to the need for the social search capability. We could also observe their active use of the new feature and change of behavior while they were using the search system. At the initial stage of our experiment, users did not show big difference in their usage of the system compared to the conventional search services but as time passes and the usage history accumulates, meaningful changes in their behavior toward the use of social navigation support features of the system were discovered.

1 Introduction

The explosive growth of the Web and its information contents addresses the need for the design of effective tools which can help people to find out proper information they need in an efficient way. Various Web search services have been developed and used so far but their quality of services in terms of user needs has been far from perfection and the problems of these services have been continuously pointed out.

Most of these Web search services are based on the traditional approach of information retrieval, which assumes that the query space and the document space are identical. However, in a real situation, especially in the new Web environment, it is not quite true. Web search service users are formulating very ambiguous and short queries unlike the experimental setting where a lot of queries were formulated and refined by domain experts and the length of them was long enough to express users' information needs. Most users are not familiar with expressing their needs in exact query terms which appears in the document space and the number of terms used for their queries are just two or three in average [8]. This situation brings the mismatch between the query space and the document space [6].

Also, most of the Web search engines adopt a "one size fits all" approach. Different users get the same set of search results if they use the same query with other users when they use search engines based on this approach. They are not personalized and are context-independent, so they can't serve different users' different needs.

Various new ideas have been developed to overcome the limitations of the traditional approach. For example, Google is exploiting link-connectivity information in addition to the traditional term based retrieval model. It applies higher weights to the documents which has more in-link counts and let them appear closer to the beginning of the retrieved result list [1]. *Social search* is another attempt to improve the traditional approach. Like link-connectivity based approach, it makes use of new features to promote the effectiveness of search results. A group of different users who share the same interest can use similar query terms for the same task and their search experiments can be shared. Based on this observation, social search approach exploits past search histories. When a user enters a query, the social search system looks up the search history of the group where the user belongs to and can provide better search results by re-ranking with the clues extracted from the search history or by providing the user with more evidences in addition to the baseline term matching retrieval systems. We have designed and implemented a social search engine for a social navigation system Knowledge Sea. In order to test users' need for this system and to find out if their behavior is different with that of conventional search systems, we conducted user surveys and transaction log analysis.

2 Background

2.1 Social navigation

Social navigation [4] research tries to explore methods for organizing users' explicit and implicit feedback in order to support information navigation in a meaningful way [2]. This approach includes two features. The first feature is to support a known social phenomenon, which means that people tend to follow the "footprints" of other people. The second important feature is self-organization, which allows social navigation systems to function with little or without manual endeavors of human administrators or experts. The well known exemplar systems based on this approach are Web auctions or Weblogs.

Jon Dron and others [5] also introduced CoFind (Collaborative Filter in N Dimensions), which structures and selects learning resources for teachers. This system was inspired by the concept of *stigmergy*. Stigmergy is a word coined by Grasse and it refers to systems employed by termites when building mounds [7]. When termites build mounds or ants form trails, they can produce mounds and trails by following their colleagues' traces in a collaborative way. These outputs can become stronger as time passes and more group members participate. They also can dissipate if a specific cause runs out and the members' participation decreases, in such a way that when food runs out, the trail to the location of the food dissipates as time passes and ants follow less after they found out no more food from there.

2.2 Knowledge Sea

Knowledge Sea is a Web-based social navigation support system. It organizes Web-based open and closed corpus C language teaching materials including online tutorials and lecture slides. In order to implement this mixed corpus based social navigation, Knowledge Sea uses a knowledge map of the domain [3] – a two-dimensional table consisting of 64 cells. It is built by self-organized map (SOM) algorithm. Semantically related keywords and documents were assigned for each cell. Contents of neighboring cells are semantically related.

Background colors of the cells indicate the popularity of the cells. As more users click and visit a cell, the background color of the cell gets darker. When they click a cell, they can see a list of documents and can choose a document from the list.

The same logic to represent popularities by color lightness is applied to the representation of documents inside each cell. Each item of the list provides two types of information, traffic and annotations. "Human-figure" icons and colors provide users with popularity information and "thumbs-up" or "thermometer" icon and colors provide users with annotation information. If a document is popular among the group where a user belongs to, the background color of the icon gets darker. The foreground color of the icon gets lighter if the user clicked the document fewer times than other group members. Just like popularity, darker background color of an annotation icon indicates there are a lot of annotations for the document. "Sticky-note," "thumbs-up," and "question-mark" icons indicate "General," "Praise," and "Question" annotations respectively. A red "thermometer" icon indicates the overall annotations are positive, and a blue icon indicates the overall annotations are negative. Therefore, users can navigate socially by referring to other users' behavior and opinions by looking up these icons and colors provided by Knowledge Sea [2].

2.3 Social search

Social search or collaborative search is an approach to promote the effectiveness of web search by relying on past search histories [6]. Smyth and others [6] implemented and tested I-SPY which is based on the concept of social search. This system is based on the observation that for specialized topic searches, the number repetition of query terms is higher than that of general topic searches. Therefore, they stored query-document frequency matrix from past search histories of the community users and re-ranked search results by looking up these query-document frequencies. They reported improvement of search results by this approach.

This study tried to implement social search capability to the existing Knowledge Sea social navigation system. Along with the browsing mode provided by Knowledge Sea, we added a search interface and let users directly search for documents they needed. Because it is based on Knowledge Sea and share the corpus and database with Knowledge Sea, the users could retrieve search results with social navigation information and make use of it.

3 System Design and Implementation

As described above, the search functionality was added to the social navigation system Knowledge Sea. In the original Knowledge Sea, users access documents through navigation using knowledge map and links between documents. Social navigation support assists the users in their navigation. With the additional search functionality, users are able access documents by entering query terms and continue their social navigation (Figure 1).

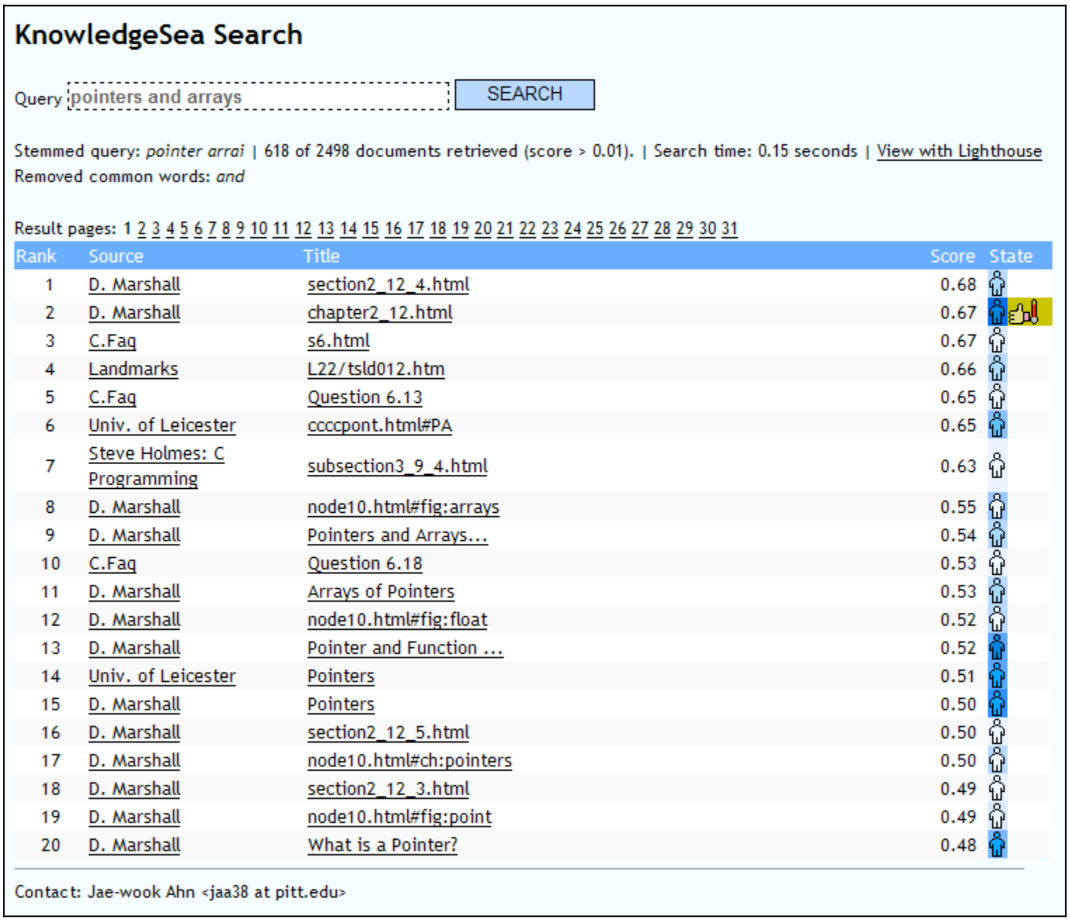

Fig. 1. Knowledge Sea search interface

3.1 Document Processing

The document collection indexed by the search system is the Knowledge Sea collection of C language educational materials. Knowledge Sea stores URL's of all documents to generate social navigation cues in runtime. The search system fetches these URL's, downloads and indexes them to make them searchable.

Terms collected from the documents are stemmed according to Porter's stemming algorithm [9]. Very common or rare terms which are stored in a stopword list were excluded. However, due to the characteristics of the document collection, which is a C language tutorial pages, some stopwords such as "if", "for", and "while" should be stored in the index because users can use them as query terms and retrieve documents containing them. Therefore, a C keyword list was constructed and they were also included in the index. The identical process is applied to query terms when users enter queries.

The terms stored in the index of the search system were weighted by their importance for each document. The weighting scheme used is TF-IDF, which means TF (Term Frequency, frequencies of terms for each document) multiplied by IDF (Inverse Document Frequency, inverse of the number of documents where a term appears). TF means how many times a given term appeared in a document and indicates the importance of the term in the document. IDF means the degree of concentration of a given term in the document corpus. Therefore, if a term appears in a small number of documents with high frequencies within them, it is more highly weighted than other terms. For queries, the same weights 1 were used for every term.

3.2 Retrieval Model

The vector space model [10] was used for representing documents and queries. Documents in the corpus and users' queries were represented as vectors. Each element in document vectors represents a term and it has TF-IDF weight. If a term appears in a document, it has the weight of term frequency in the document multiplied by inverse document frequency in the corpus. Elements in query vectors also represent terms and they were represented as binary, that is, the weight of the term is 1 if it appears in the query or zero otherwise.

By comparing a query and document vectors, we can produce a list of documents, which are similar to the user query. They are ranked and ordered by their similarity to the query. We use traditional *cosine similarity* between query and document vectors. Cosine similarity coefficient was calculated with equation 1, where x and y represents query and document vectors. Documents with similarity values above 0.01 were displayed (20 per page).

$$Sim(x,y) = \frac{\sum_i x_i y_i}{\sqrt{\sum_i x_i^2}\sqrt{\sum_i y_i^2}} \quad \mathbf{(1)}$$

3.3 Implementation of Social Search

In addition to the conventional features of this search engine, social navigational features are supported. In response to users' queries, a set of documents sorted by their similarity with the query are retrieved and ranks, document titles, sources of the documents, and similarity scores are displayed for one record. Along with this conventional information, social navigation information is also displayed with proper icons and different foreground/background colors at the end of each record.

The search service shares traffic and annotation database with Knowledge Sea. It retrieves social navigation information from the database and shows it along with search results. When a user clicks any record and views its contents, a document display window of Knowledge Sea is opened. Page visit information in database is automatically updated and users can make annotations just like when they annotate using Knowledge Sea.

This system supports two types of social navigation information: *navigation traffic* (how many times users selected a document) and *annotation* (annotations made by users to a document). Traffic includes user traffic and group traffic. User traffic means the traffic of the current user who is using the system and group traffic means the traffic of other users of the group the current user belongs to. Annotations include “Praise,” “General,” and “Question” types and it also represents whether they are positive or negative. Traffic part of social navigation makes use of human-shaped icons. The blue background colors of the icon represent the group traffic. As group members select and view the corresponding document, the group traffic increases and the blue background gets darker. The foreground colors of the icon means user traffic. If the user has viewed the document more than the average user of the group, the icon is darker than the background. If the user has viewed the document less than the average, the icon is lighter than the background. Figure 2 shows two different records with same similarity scores. Even though their similarity with a given query is identical, the traffic information is different. We can easily see that group users have visited the second record more times than the first record by its darker background color. We can also see that the current user visited these records as frequently as other group users because the foreground and background colors of these records are identical.

Fig. 2. Traffic information

The number of annotations is represented as the darkness of background colors. As users annotate a document more, the yellow background color of the annotation part gets darker. For three different types of “General,” “Praise,” and “Question” annotations, “sticky-note” “thumbs-up,” and “question-mark” icons were used respectively. In order to represent whether the overall annotations for a document are positive or negative, “thermometer” icons were used. For positive annotations, red colored “thermometer” icons were used and for negative annotations, blue colored “thermometer” icons were used. From the example record in Figure 3, we can find out that it has a lot of annotations (darker background), “Praise” annotations (“thumbs-up” icon), and the annotations are positive (red “thermometer icon”).

Fig. 3. Annotation information

4 Research Design

4.1 Research questions and hypotheses

This study attempts to answer the following questions.

1. Do users agree with the need for the search functionality of social navigation?
2. Do they consider the social navigation information more important than document ranks when they select a document in the list of search results?

The first question is about whether the real users will need the new search capability with social navigation support along with the baseline Knowledge Sea system. The other one is related to the situation when the users retrieved documents using the social search. The search result provides users with two types of different information at the same time, conventional similarity ranks and social navigation information. Therefore, users should select documents on the basis of these information and we are interested in the type of information users depend on more. Based on these questions, we have established two hypotheses.

1. Users will need the social search capability and will use it meaningful times.
2. Users will actively select documents with higher social navigation scores. Especially, they will select lower ranked documents (appeared lower part of the retrieved results) if the documents have high group traffic and/or positive annotations.

4.2 Data collection

To answer these questions, user survey and usage log analysis of the system were conducted. For the survey, following questions were asked to the students of INFSCI 0012 Introduction to Programming class at the School of Information Sciences, University of Pittsburgh.

1. "The availability of search interface in Knowledge Sea was important".
2. "Unlike traditional search engines that return the list of results ordered by relevance, Knowledge Sea search also shows you using standard color metaphors how many visits you and your group made to the pages found. This feature was useful in deciding which pages in the list of search results to visit."

We have kept the transaction records of how users behave when they browse, search, and select documents using this system. The transaction logger keeps track of users' navigational behavior. Therefore, we can find out whether users used browsing or searching mode to select an educational document from this data. Also, we can extract the similarity rank and the social navigation score of the documents when users selected and viewed them.

5 Analysis of Results

5.1 User Survey

Nine students answered the survey questions. The results are shown in Figure 4. For the first question asking about the need for the search interface, about 88.9% of the students agreed the need for the search capability for social navigation system. 11.1% of them were neutral, and no student expressed disagreement with this need. For the question asking the need for the social navigational functionality of retrieved results, 77.8% of the students agreed, 11.1% of them were neutral, and 11.1% disagreed.

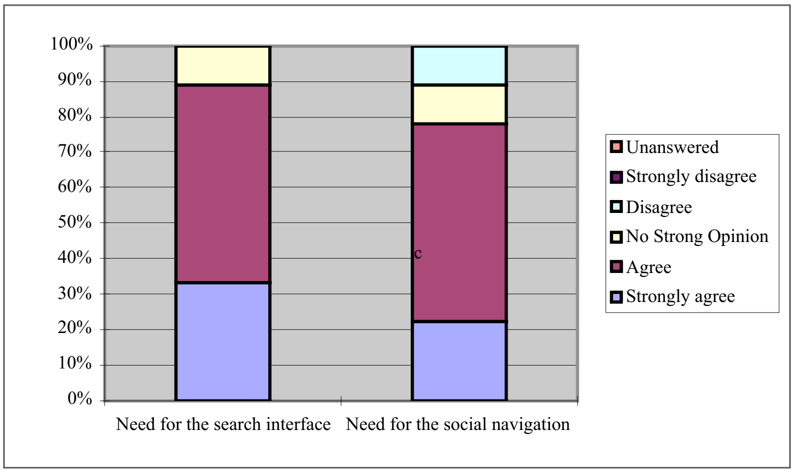

Fig. 4. Students' attitude to the need for the search interface and the social navigation

5.2 Transaction log analysis

First, the transaction log for two months (from 10/19/2004 to 12/18/2004) was analyzed. This data contains the frequencies of each mode users had used before they finally located and opened a document. Users can choose a cell and browse using Knowledge Sea's baseline system, or directly search for relevant documents by entering queries to locate relevant materials they need. With this data, we can find out users' preference on each mode before they reached educational document. The result is summarized in Table 1. The most frequently used mode was browsing. Map mode and searching mode were used about 1.5 and 4 times less than the browsing mode respectively.

Map	Browsing	Searching	Total
299 (36.2%)	423 (51.1%)	105 (12.7%)	827

Table 1. Number of times used for each navigation mode

Part of the transaction log data contains more information about users' behavior. For one month (from 11/16/2004 to 12/18/04), we have recorded data which can be used to analyze social search activities. The additional information includes document rank calculated by similarity, document ID, query string, number of accesses for the corresponding document made by the user himself and other users, the number of annotations, and annotation types. By analyzing this data, we can see if users preferred conventional rank information provided by the search engine or the popularity and annotations of other group users.

53 document selections were recorded within this time period. Each selection was made on the basis of information available to the users before they opened a specific document. That includes rank, social navigation cues (if present), and title. Table 2 shows the average rank and count of selected documents for two groups. One group is documents with social navigation cues and the other is without such information. This

result is corresponding to our expectation in a part and not in part. As we have expected, the users selected documents with social cues slightly more frequently than documents without social information (29 to 24) even though most of the documents in the result list were not annotated. Together with the user survey, this data provides some support for our first hypothesis that users will need the social search capability and will use it meaningful times.

We also have expected that the users would select documents with lower ranks if they were popular and/or annotated. However, the overall average rank in Table 2 shows that users still preferred higher ranked documents even though the documents were emphasized by social navigation information.

	With social navigation	Without social navigation
Average rank	6.48	8.54
Selection count	29	24

Table 2. Average rank of documents with and without social navigation information. Note that the higher rank corresponds to lower numbers.

Table 3 shows the number of documents viewed per query by popularity and a n-notations. Users can distinguish popular documents among group users with higher group traffic by looking at the background colors of the "human-figure" icons and they can also distinguish positively annotated documents by looking at the colors of the "thermometer" icons. They selected and viewed about 1.3 times more documents when they retrieved results which include documents other group members had viewed before them. For positive social annotation, they selected and viewed 2.4 times more documents among the retrieved results than they retrieved results without positive annotations. To summarize, users tried more items among their retrieved set when they saw higher traffic items or positively annotated item

	Average	Total	# of queries
With group traffic cues	2.69	35	13
Without group traffic cues	2	18	9
With positive annotations	4.5	18	18
Without positive annotations	1.94	35	4

Table 3. Number of documents viewed per each query split by presence of traffic and annotation social cues

The data above shows that the retrieved documents with social navigational information were popular among the users. However, in terms of the average rank of the selected documents, the average rank score of the documents with social navigation information was higher (numerically smaller) than the others. This does not correspond to our expectations that the users will choose lower rank documents if they have higher traffic or positive annotations.

For more careful evaluation of social navigation support in the list of search results, we considered two factors: which pages were selected and how much time the student spent reading each page after it was selected in the list. Since students do not see the content of the page while looking at search results, the fact of page selection reflects the perceived relevance of page for the students that is formed on the basis of page title, rank in the list, and possible social cues. In contrast, time spent reading (TSR) the page reflects the "true relevance" of the page – it's usefulness for the student. The more clicks were made on links of a specific category, the higher is the perceived value of this category. The larger is TSR pages behind the links of a category, the higher is the "true relevance" of this category.

To evaluate the "true relevance" of pages with low and high group traffic we compared time spent reading a page for pages with high group traffic and low group traffic. Similarly, we assessed TSR for pages with high rank versus pages with low rank. To evaluate the perceived relevance for these categories, we compared the number of accessed documents in each category. We hypothesize that pages with social cues will have higher perceived relevance (because the cues attract students attention) and higher true relevance (because they are "approved" by the group as a whole). In contrast, we expected high-ranked pages to have higher perceived relevance and lower true relevance. For evaluation we looked at median TSR over all selected pages from search result. We used median to discard too long or too short TSR. The group traffic represents the number of clicks before the page is being chosen from the search result. We consider pages with 3 or more clicks as pages with high group traffic since 3 clicks make the background clearly dark. Also we consider the first three search results as "high rank".

Our result is consistent with our hypotheses. As shown in the table 4, high-ranked links attract students – almost 1/2 of all clicks are done one the top three links. Yet, as table 5 shows, this attraction is often misleading: students realize quickly that the page is not relevant and spent little time reading it. Overall, as we expected, high rank is a very poor predictor of how interesting and relevant the page really is – as measured by very low average TSR (13 sec). At the same time, high traffic annotation, while not attracting student attention as much as high rank (17 vs. 20) is a much better predictor of relevance with average TSR 31 sec. Interesting is that the best predictor of relevance is a combination of high rank and high traffic – with TSR 56.5 second. While high rank/high traffic pages turned out to be the best for students, they are less frequently visited. So, while the students like traffic-based annotation and it does help to get to relevant pages, they still do not trust it as much as it deserves according to its performance.

Number of accessed documents			
	Low rank	High Rank	Total
Low group traffic	19	12	31
High group traffic	9	8	17
Total	28	20	48

Table 4. Document distribution by rank and group traffic

Median TSR			
	Low rank	High Rank	Total
Low group traffic	50	8	25
High group traffic	21	56.5	31
Total	41.5	13	26.5

Table 5. Effect of social navigation on accessing from search result by TSR (Time Spent Reading)

6 Conclusions

In this study, we added a social search capability to a social adaptive navigation system Knowledge Sea and tested its usability. We implemented a service, which shares traffic and annotation information with Knowledge Sea and let users make use of social navigation features within our social search system. We expect this new feature will improve the effectiveness of our system and the social search capability will overcome the limitations of the traditional Web search services.

By implementing this system, we tried to find out if users really needed the social search capability and if they would show behavior, which is different from when they use traditional search services. Users tend to select documents displayed in the upper part of the retrieved result set. However, with additional clues like group user traffic and positive annotations implemented in our system, we could expect a change in users' document selection behavior.

Therefore, we established two hypotheses. First, users will need the social search capability and will use it meaningful times. Second, users will actively select documents with higher social navigation scores. To test these hypotheses, we conducted a survey and analyzed the transaction log. According to the survey, very high number of users agreed with the importance of search interface and the usefulness of the social navigation support for the search interface. We were also able to find out from the transaction log that the search interface was used in a significant number of times even though the existing map and browsing system of Knowledge Sea were used more often.

To analyze users' document selection behavior in terms of social navigation information, we observed selection counts and the ranks of the selections and tried to find out if users were willing to select and view documents with higher group user traffic and positive annotations. Our assumption was that users would actively select and view popular and positively annotated documents and they would select such documents even though the documents had lower rank score and appeared at the lower part of the retrieved result set. However, overall average rank of documents with social navigation information was higher than those with such information unlike our expectation. Therefore, we tried to see how users' behavior change as time passes and found out users tended to select lower rank documents as their usage history accumulates.

We also analyzed TSR (Times Spent Reading), which means how much time users spent for reading pages according to their characteristics like ranks and group traffics.

Results show that users spent much more time reading documents with lower ranks and higher group traffics. It also supports the relationship between ranks and group traffic of documents and users' choices of those documents. From these results, we can conclude that users tend to select documents with lower ranks if they are provided with additional social navigation information like group traffic.

We have also found out that users' inactive selection behavior in the earlier stage of the experiment was caused from the Cold-Start-Problem, which happens at the earlier stage of social navigation systems when enough user history information is not collected. Therefore, we can expect users to exploit popular and positively rated items more and more actively with a system that supports social search and social navigation features as the accumulation of social navigation information increases.

Conventional rank information for information retrieval system is not enough for support users to select relevant documents. With the help of group users' tacit or explicit evaluation on that information, user can more effectively complete their task to find out documents they really need.

7 Acknowledgement

This material is based upon work supported by the National Science Foundation under Grant No. 0447083.

8 References

1. Brin S. and Page L.: The Anatomy of A Large-Scale Web Search Engine. In Proceedings of the Seventh International World-Wide Web Conference (1998)
2. Brusilovsky, P., Chavan, G., and Farzan, R.: Social adaptive navigation support for open corpus electronic textbooks. In: P. De Bra (ed.) Proceedings of Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2004), Eindhoven, the Netherlands (2004)
3. Brusilovsky, P., and Rizzo, R.: Using maps and landmarks for navigation between closed and open corpus hyperspace in Web-based education. The New Review of Hypermedia and Multimedia 9 (2002) 59-82
4. Dieberger, A., Dourish, P., Höök, K., Resnick, P., and Wexelblat, A.: Social Navigation: Techniques for Building More Usable Systems. Interactions 7(6) (2000) 36-45
5. Dron, J., Boyne, C., and Mitchell, R.: Footpaths in the Stuff Swamp. In: Fowler, W. and Hasebrook, J. (eds.): Proc. of WebNet'2001, World Conference of the WWW and Internet, Orlando, FL, AACE (2001) 323-328
6. Freyne J., Smyth B.: An Experiment in Social Search. In: P. De Bra (ed.) Proceedings of Third International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH'2004), Eindhoven, the Netherlands (2004)
7. Heylighen F.: Collective Intelligence and its Implementation on the Web: algorithm to develop a collective mental map. Computational and Mathematical Theory of Organizations 5(3) (1999) 253-280
8. Lawrence S. and Giles C.L.: Context and Page Analysis for Improved Web Search. IEEE Internet Computing, July-August (1998) 38–46
9. Porter, M.F.: An algorithm for suffix stripping, Program, 14(3), (1980) 130-137
10. Salton, G.: Automatic Text Processing, Addison- Wesley Publishing Co., Reading, MA, (1989) 313-326

Using a Domain Ontology to Mediate between a User Model and Domain Applications

Charles Callaway and Tsvi Kuflik

[1] University of Edinburgh, 2 Buccleuch Place
Edinburgh, Scotland EH9 1SA
ccallawa@inf.ed.ac.uk
http://homepages.inf.ed.ac.uk/ccallawa/
[2] MIS department, University of Haifa, Mount Carmel, Haifa 31905, Israel
tsvikak@is.haifa.ac.il
http://mis.hevra.haifa.ac.il/∼tsvikak/Home.htm

Abstract. User modeling data in dynamic, personal and adaptable systems is usually collected immediately before system interaction with a questionnaire, or during application execution when users' choices are recorded and analyzed. This data is then typically used to intelligently adapt the system's output, hopefully improving the user's interaction in some measurable way. When coupled with knowledge-based applications such as intelligent information presentations or tutoring systems, this user model may be mapped onto the system's knowledge base as an *overlay* that may describe what domain material has been experienced by the user or to adaptively encode a progression of topics to be presented next. We present a case study in the museum domain, where an adaptive hypermedia mobile presentation system creates a user model for its own use, and subsequently a post-visit report generation system modifies the data in the model to produce a personalized summary of the entire museum visit. We describe how one component, the interest model, is seeded by the knowledge-based user modeling data collected in the initial mobile phase, and is then expanded via inference over the existing domain ontology during the second phase of report generation.

1 Introduction

The need for information presentation systems to automatically adapt themselves to their users is recognized in many application areas, among them visitor guides for tourists like CyberGuide [9] or Deep Map [6], and museum visitor guides as developed for projects such as Hips [12]. In order to adapt a system to individual users, there is a need to identify each user's needs and to model the user in order to guide the adaptation process. Such a user model should provide the information needed by the specific application, and hence usually contains only relevant application-related data.

Information about users can be collected in two ways: explicitly, by asking the user specific questions in order to provide the relevant information (an approach adapted by the GUIDE context aware tourist electronic guide which asked its users to provide some personal details such as their name, interests and preferred reading language [5]) or implicitly, by monitoring users' behavior and inferring a user model based on a number of observations, an approach taken for example by researchers for the HIPS system in the non-intrusive version of their museum visitor's guide [10]. A major drawback of explicit user modeling is the need for users' active involvement, an effort users typically seek to avoid given the need for asking a number of potentially intrusive, private questions. Implicit approaches are also troublesome given that inferring user interest and knowledge from their behavior is a highly uncertain task.

Using stereotypes – a representation of clustered groups of users – is one way to partially overcome these limitations: users are requested to provide a small amount of personal (but not identifying) information that allows the system to assign them to a stereotypical group from which relevant user details can be inherited. Such an approach was taken in the INTRIGUE project [2], a mobile tourist guide for the city of Turin. INTRIGUE recommended destinations and itineraries for family groups, allowing for various points of view such as historical period and artistic themes. It elicited user desires with direct questions, storing demographic and background data for each person in the group, and then using a probabilistic user model to predict their joint interests.

Like other guide domains, the museum environment [17] is a challenging environment for user modeling. First, extensive and detailed information about the museum exhibits is needed in order to provide the relevant knowledge space to match the wide array of possible presentations generated for particular users on their individual paths through the physical space. In addition, there is very little information (if at all) about the visitors entering the museum, and it is impossible in a real museum environment to ask them to explicitly provide all the information required to create an extensive user model. Hence non-intrusive user modeling that can still yield adaptive, personalized results is required.

Specific challenges include user modeling, as described by Serini and Straparava [19] in the HIPPIE project, and has been the subject of a wide range of research efforts with well-developed user models ranging from kiosk-based clients [1] to mobile PDAs [18]. HIPPIE represented a user by a user model that took into account the user's personal data (such as age, job and more), his level of domain knowledge and his interests. The unique setting and the complex nature of the museum together with the highly varied characteristics of the visitors led researchers to suggest very personal support would be preferable to the stereotype based approach, which was confirmed in subsequent highly detailed but unimplemented research on the HYPERAUDIO system [13].

A significant advantage of the deep representational level inherent in knowledge bases is the availability of a taxonomy of concepts (ontology) for use in semantic processing, for instance to gauge similarity between two domain concepts or to record exactly what information has already been presented to the user. This is true in other domains needing user modeling beyond museum or tour guides such as intelligent tutoring systems [11] and e-learning [7].

We describe two of our museum applications in the PEACH project which both share a common knowledge base, but employ a user model in different ways. Our KB supports typical functions such as reasoning for overlays and for determining sets of subsequent potential choices for the user to explore next. In addition, it also supports a sophisticated natural language generation interface which immediately after a museum tour allows us to create printed or emailed reports for museum visitors, consisting of a record of their individual interests and recommendations for followup learning, which they can then read at home or online after their museum visit. The report generator takes the original user model created implicitly during the visitor's tour, makes extensive use of the existing KB ontology to infer a personalized interest model, and creates the report detailing what he or she might like to see during their next visit to that museum or other similar nearby museums.

2 The PEACH Domain, Mobile Guide, and Report Generator

A visit to a museum in the context of the PEACH project consists of two separate phases: an immersive visit through the museum accompanied by a PDA which transmits interface events and receives animated, interactive presentations for the visitor [16], and a natural language report generator [3] which delivers a personalized electronic or paper report describing the visitor's trip through the museum along with additional information that the accumulated interest model predicts that that particular visitor would like to see. The user model is formed by the individual path the visitor takes through the museum and the choices for information on new objects they made via the PDA interface [8].

As an application domain we have chosen the *Cycle of the Months* of the Torre Aquila at the Buonconsiglio Castle in the city of Trento, Italy. This work is composed of eleven side-by-side frescos (measuring on average 2 meters wide and 3 meters high each, and representing a particular calendar month, such as January in Figure 1) painted during the 1400s and illustrates in great detail and complexity the activities of medieval aristocrats and peasants of Trento throughout a full year. Each fresco presents several scenes that were typical of daily life in that month, such as aristocratic, plebeian, and religious activities. The frescos are highly detailed, depicting architectural details, clothing and tools of the era, providing an environment rich in similarities and differences which can be exploited by the user model and domain knowledge base.

The underlying knowledge base contains around 1500 domain concepts for each visible object in a fresco (people, animals, buildings, etc.), properties of those objects (color, size, history, spatial position, etc.), active and stative relations (jousting, building, sitting, etc.), plus a larger number of generic concepts necessary for generating explanations with the report generator. Additionally, these concepts are organized under an *ontology* that separates concepts of unlike types and allows an inference engine to determine which concepts are similar, and thus might be of interest to visitors. The ontology is coded directly into the

Fig. 1. The January fresco of the Torre Aquila in Trento, Italy

KB as taxonomic relations between concepts of varying abstraction. The KB was built completely by hand, requiring around three months of effort.

2.1 The Mobile Guide

The interactive museum tour takes place with an adaptive mobile museum visitors guide developed within the PEACH project. The electronic guide provides museum visitors with dynamically generated personalized presentations relevant to the exhibits they see in the Torre Aquila. They include details about specific scenes found in the frescos, presented from a certain perspective (such as artistic or historical). The presentations are based on detailed information that is drawn from the system's knowledge base, and is selected based on user preferences as inferred by the system [15]. Presentations are given to the visitors on a choice of devices - a PDA that the visitor carries, or a desktop with better displaying capabilities, if the visitor is near such a device when a relevant presentation is prepared. The personalization is based on a set of features representing information continuously gathered about the visitor during the museum visit, including:

- *Spatial information*: whether the user has already been in this area before, is facing an artwork, or she/he is spatially nearby some exhibit;
- *Interests*: whether the user is interested in a specific artifact;

– *Discourse history*: what particular shots the user has already seen, or if she/he has already watched a presentation for an artwork;

– *Device*: if the user is visiting a museum provided with a PDA, or she/he is looking for information on the web with a desktop PC;

The PEACH dynamic user modeler works in a "non-intrusive" manner. Hence there is no information about the visitor when starting each visit, and as a result, the model is built solely by observing the visitor's behavior. From the beginning, the user is tracked by recording their positions (in terms of the visited exhibits), the time spent at each position, and the details about the presentation delivered (the main theme, the global perspective and peripheral concepts included). User interests are defined in terms of domain concepts, which are associated with individual presentations. These concepts provide a description of the content of the presentation, thus representing its theme. The concepts that are associated with the presentations and used for modeling user interests are drawn from a domain knowledge base that is primarily designed for natural language generation for visit summary reports. Since there is no prior knowledge about the user, the knowledge base and the concepts associated with the individual presentation are the only source of information for user preferences with respect to the exhibits visited and presentations delivered in the current museum visit.

The information described above is continuously being gathered and the level of user interest inferred by the user model working in a "non-intrusive" manner. User level of interest is inferred by monitoring each user's explicit and implicit feedback. Explicit user feedback is given by pressing a "More" button (to indicate a positive reaction) or an "Enough" button (for negative reaction) and implicit feedback, in the form of presentation completed without interruption (positive reaction). These are used to infer user interest in the various concepts presented by the presentations that the visitor sees. Using an inference mechanism that follows ontological links in the knowledge base, user interests are propagated from the specific concepts associated with the presentations to more abstract, related concepts (*e.g.*, interest in a concept "knight" is propagated to the more abstract concept "aristocracy").

Explicit feedback has a higher priority than implicit feedback in the sense that explicit feedback is more reliable so it drives an immediate change in level of interest in the concepts associated with the delivered presentation, while implicit feedback requires accumulation of evidence for every concept (several implicit responses) before changing a visitor's interest level in that given concept. In addition to the level of interest, a "certainty factor" is used as a way of representing the semantic distance between the original concept where the inference started and the current, inferred concept. Concepts which are part of presentations are seeded with an initial neutral value – "interested a little". Later interactions change the model of the visitor's interests in the various concepts using a qualitative 5-level scale (for more details, see [8]).

2.2 The Report Generator

Supporting adaptive, intelligent presentation generation requires recording events during the visit, updating information about user interests, and choosing new presentations based on these inferred user interests, as described above. Generating a subsequent report about that visit instead requires a different perspective on the type of information presented and the types of inferences used because we want to (1) explicitly describe in detail exhibits that seemed to be of special interest to the visitor, (2) compare and contrast details in order to increase their understanding of what they have seen, (3) describe museum exhibits the visitor didn't see in addition to those seen (in order to trigger interest for future visits), and (4) recommend a number of exhibits both in this and in other museums so as to enrich the overall user experience.

To create the text of the report, the report generator combines a text planner that determines the most relevant information to put into the description along with its coherent rhetorical organization, and a deep syntactic NLG system that creates the actual text read by the visitor. The text planner accesses the user model, ensuring that the resulting text will be personalized, and makes extensive use of the knowledge-based ontology and ontological inference mechanism described below to decide what were the favorite exhibits seen by the visitor as well as what exhibits might have been favorites if the visitor had had enough time to see them.

For adaptive generation that is highly personalized for a particular museum-goer's visit, it is important to ensure a high amount of variation in the resulting text. To achieve such variability, the text planner queries the user model to get the log of the user interactions. For instance, to *sequentially* describe what the visitor saw, the text planner extracts an ordered list of visited artworks and accesses the knowledge base to get a shallow description of the main contents of each artwork to be included in the summary.

Alternatively, the text planner can retrieve a list of ranked topics from the inferred interest model. Thus the corresponding *thematic* report may consist of a series of paragraphs describing the top items in the interest model. To prevent excess repetition of similar exhibit types (for instance, where the entire text is about the knights, lords, and ladies in the frescos), we cluster semantically related items in the interest model, making extensive use of the ontology to inform us which items are conceptually related to others. The text planner also includes additional details from the knowledge base based on perspectives the user seemed globally interested in. For example, if the visitor spent a lot of time watching and requesting information on castles, churches and buildings in the frescos, the ontologically informed heuristic assumes the user is interested in architecture and thus includes architectural facts from the knowledge base, such as the name of the builder or its particular building style.

The text of the report is created using the language-independent StoryBook deep NLG system [4], which handles low-level language issues, such as sentence subjects, verbs, pronouns, morphology, etc. Deep NLG has advantages that make it useful in generating extended reports in a museum context: the text can be

Visit Log / User Model	Initial Interest Model	Extended Interest Model
Moved in front of January for 223.0s	HUNTERS +	ARISTOCRACY +
Started January-Fresco Overview	HUNTING-DOGS +	ARIST.-ACTIVITIES +
Completed Presentation on HUNTERS	BADGERS +	ANIMALS +
Completed Presentation on CASTLE	SNOW +	WINTER +
Stopped Presentation on CASTLE-WALLS	CASTLE +	ARCHITECTURE -
	CASTLE-WALLS -	BATTLE -
Moved in front of February for 192.0s	TOURNAMENT +	ARIST.-ACTIVITIES +
Started February-Fresco Overview	KNIGHTS +	ARISTOCRACY +
Completed Presentation on TOURNAMENT	LANCES +	WEAPONS +
Completed Presentation on KNIGHTS	HORSES +	ANIMALS +
Stopped Presentation on SPECTATORS	SQUIRES +	SOCIAL-ROLES +
Completed Presentation on BLACKSMITH	SPECTATORS -	ARISTOCRACY -

Sequential, Non-Adaptive Visit Paragraph	Thematic, Adaptive Visit Paragraph
... You first went to see the January fresco which contains many scenes of winter activities. The main theme of this fresco is a snowball fight between a group of nobles in the bottom panel. Two hunters are leading their dogs to search for badgers, while a lord is cutting roses in his castle garden ...	... Your favorite theme was the activities that the nobles engaged in during their daily lives. For instance, the hunt in the snow in January, the knights engaged in the tournament in February, and September's hawk hunts captured your attention for a large part of your visit ...

Table 1. Inferring an interest model from a visit log.

in multiple languages, produced in high-quality prose, provide for automatic variation at the syntactic and lexical levels, and contain integrated markup (such as HTML, or prosody for text-to-speech). Other NLG systems have had user modeling components, such as the STOP [14] report generation system; though our implementation is meant to work in multiple languages and does not require any explicit information from the user. Importantly for the production of printed, color reports that will be read by museum visitors, the NLG system allows the introduction of HTML markup into the text at the syntactic level and thus can produce the report as a web page which can also be emailed [3], and the report includes an image of each artwork the visitor was interested in.

3 Inferring the Interest Model using a Taxonomy for Presentation Generation and Report Generation

Between the completion of the interactive tour and the creation and printing of the report as the visitor leaves, the user model that results from the tour must be extended to allow for inference of the interest model that covers more than the exhibits seen by the visitor. Otherwise, the visitor would receive a report that is merely a copy of the visit log, describing the sequence of exhibits visited.

For instance, imagine a visitor who carries their PDA in front of the January fresco, as described in Table 1, and watches a series of presentations about aristocrats enjoying leisurely winter activities (Figure 1). The visitor watches entire presentations about these topics, but interrupts the PDA when it begins to describe the architectural details in the same scene. The visitor then moves on to the February fresco, and again listens patiently to presentations about aristocratic activities in addition to those of the lower social classes. The visit log collects this data from the mobile presentation system, and integrates it with the knowledge base, producing an initial interest model.

From the joint knowledge that the visitor enjoyed a hunting scene in the January fresco and a tournament in the February fresco, we can generalize to infer that the user is interested in winter aristocratic leisure activities. Integral to this inference is ontological knowledge as well as other types of relations that connect concepts into a large semantic net. The knowledge base records information about each scene being presented, for instance that the presentation on hunters includes the hunting dogs, a badger, the hunters themselves, etc. Later, after inference, the text planner may additionally cluster multiple similar conceptual interests to produce abstract topics such as "aristocrats" rather than "knights" and "ladies" individually.

If well-defined clusters can be created for the top-rated concepts at the completion of inference for the interest model, the text planner then writes a report organized thematically and centered on the top clusters in the list [3]. Otherwise, the system chooses a sequential report describing exhibits in order, pulling details from the interest model. In our experience, a thematic organization of the report is superior to sequential methods of organization. In either case, adaptive variation in the report text is ensured at the organizational level, depending on the visitor's path through the museum and their requests for further details. Finding clusters is also important for report generation because without them the final report may consist of redundant sequences of text talking about similar interests without providing any contrasting or unifying information.

4 Discussion

In our application, a single user model supports two different tasks: online presentation generation and creating personalized summary reports. On one hand, the user model supporting adaptive presentation generation requires recording events during the visit and abstract information about user interests, which are more general by nature than the specific concepts depicted by the presentation. For instance, the specific jousting activity of knights may be a concept associated with a presentation, but for user modeling purpose the more abstract concept of aristocratic activity is what really matters. On the other hand, the report generator requires much more detailed information than is available in such interest models; the required knowledge should include details such as the number of jousting knights in the scene, their weapons, clothes, relative position to each other and so on in order to support potentially all possible reports from every visitor perspective. To provide this level of detail for an application, we must overcome a gap between the standard functions a user model supports for abstract user modeling and the domain-specific expectations of the application.

This gap can be seen in what was initially available to report generation: an unordered or semi-ordered list of interests extracted from the knowledge base with associated discrete interest annotations. The list included the specific concepts and related, more abstract concepts. For dynamic presentation generation, all that is needed is a list of concepts and a level of user interest in each and every one of them, so that the user model can respond to queries such as what is the level of interest the user has in some concept X (in addition to queries like whether the user has seen a concept X or did the visitor visit some exhibit X). Such a list is ill-suited for direct processing by a report generator, which needs rhetorical and discourse motivations to produce text. Given the list-like nature of the interest model, driving the text planner thus required a number of basic list functions whose parameters include the semi-ordered topic list, the knowledge base, and fundamental user parameters. However, standard implemented user models do not support these functions:

- *Filtering*: Removing or retaining particular interests that satisfy a filter condition, such as all artwork elements containing animals or farm implements.
- *Clustering*: Grouping similar interests under more abstract hierarchies (which may need to be constructed on the fly), such as aristocrats from a set of interests including lords, ladies, and knights, to avoid repetition in the report.
- *Sorting*: Placing a series of interests in some logical order, so that the report doesn't result in a sense of random order.
- *Splitting*: Separating similar items into groups depending on an external element, such as when they are distributed across adjacent artworks, and especially for incorporating rhetorical effects like comparison and contrast.
- *Searching*: Looking through the knowledge base for items similar to a given interest, which can be used to populate the text with additional details.

Such services are required for the system to group together the individual concepts in a semantically meaningful way. Such a grouping lets the system focus on concepts that were the most interesting for the visitor and for later elaborating specific concepts of interest by querying the knowledge base for more details.

5 Conclusions

Specific implementations impose specific requirements on user modeling. In this paper, we described a method to bridge the gap between an abstract user model needed for a dynamic museum guide system and a detail-centered interest model needed for a report generator. The domain knowledge base served as a foundation for this bridge, allowing implicit visitor behavior to determine an adaptive, personalized report of the visitor's experience.

References

1. Calder, J., A.C. Melengoglou, C. Callaway, E. Not, F. Pianesi, I. Androutsopoulos, C.D. Spyropoulos, G. Xydas, G. Kouroupetroglou and M. Roussou: Multilingual Personalized Information Objects. Multimodal Intelligent Information Presentation. O. Stock and M. Zancanaro (eds.), Series: Text, Speech and Language Technology, Vol. 27, Kluwer Academic Publishers, 2005.

2. Ardissono, L., A. Goy, G. Petrone, M. Segnan and P. Torasso: INTRIGUE: Personalized Recommendation of Tourist Attractions for Desktop and Handset Devices. *Applied Artificial Intelligence* 17(**8-9**):687–714, 2003.
3. Callaway, C., T. Kuflik, E. Not, A. Novello, O. Stock and M. Zancanaro: Personal Reporting of a Museum Visit as an Entry-point to Future Cultural Experience. IUI 2005.
4. Callaway, Charles and James Lester: Narrative Prose Generation. *Artificial Intelligence* 139(**2**):213–252, 2002
5. Cheverst, K., N. Davies, K. Mitchell, A. Friday and C. Efstratiou. Developing a Context-aware Electronic Tourist Guide: Some Issues and Experiences. Proceedings of the SIGCHI conference on Human factors in computing systems, The Hague, The Netherlands, pp. 17–24, 2000.
6. Fink, J. and A. Kobsa: User Modeling in Personalized City Tours. *Artificial Intelligence Review* 18(**1**):33–74, 2002.
7. Kay, J., and A. Lum: Ontologies for Scrutable Student Modelling in Adaptive E-Learning. Proceedings of the Adaptive Hypermedia and Adaptive Web-Based Systems Workshop on Semantic Web for E-Learning, August 2004.
8. Kuflik, T., C. Callaway, O. Stock and M. Zancanaro: Non-Intrusive User Modeling for a Multimedia Museum Visitors Guide System. Submitted to UM 2005.
9. Long, S., R. Kooper, G. Abowd and C. Atkinson: Rapid Prototyping of Mobile Context-Aware Applications: The Cyberguide Case Study. Proceedings of the 2nd ACM International Conference on Mobile Computing and Networking, 1996.
10. Marti, P., A. Rizzo, L. Petroni, G. Tozzi, and M. Diligenti: Adapting the museum: A non-intrusive user modeling approach. Proceedings of the Seventh International Conference on User Modeling, pp. 311–314, 1999..
11. Mizoguchi, R. and J. Bourdeau. Using ontological engineering to overcome common AI-ED problems. *International Journal of Artificial Intelligence in Education* 11:107–121, 2000.
12. Not, E., M. Sarini, O. Stock, C. Strapparava and M. Zancanaro: Information Adaptation for Physical Hypernavigation. In Proceedings of the I3 Annual Conference, Nyborg, 1998.
13. Petrelli, D., A. De Angeli and G. Convertino: A User-Centered Approach to User Modelling. Proceedings of the Seventh International Conference on User Modeling, pp. 255–264, 1999.
14. Reiter, E., R. Robertson and L. Osman: Lessons from a Failure: Generating Tailored Smoking Cessation Letters. *Artificial Intelligence* 144:41–58, 2003.
15. Rocchi, C. and M. Zancanaro M: Rhetorical Patterns for Adaptive Video Documentaries. Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems AH-2004, Eindhoven, 324–327, 2004.
16. Rocchi, C., O. Stock, M. Zancanaro, M. Kruppa and A. Krueger: The Museum Visit: Generating Seamless Personalized Presentations on Multiple Devices. Proceedings of IUI-2004, Madeira, 2004.
17. Stock, Oliviero: Language-Based Interfaces and Their Application for Cultural Tourism. *AI Magazine* 22(**1**):85–97, 2001.
18. Stock, Oliviero and Massimo Zancanaro: Intelligent Interactive Information Presentation for Cultural Tourism. Invited talk at the International Workshop on Natural, Intelligent and Effective Interaction in Multimodal Dialogue Systems. Copenhagen, Denmark. June, 2002.
19. Sarini, Marcello and Carlo Strapparava: Building a User Model for a Museum Exploration and Information-Providing Adaptive System. Proceedings of the 2nd Workshop on Adaptive Hypertext and Hypermedia, Pittsburgh, PA, June, 1998.

A Meta Search Engine for User Adaptive Information Retrieval Interfaces for Desktop and Mobile Devices

Ernesto William De Luca and Andreas Nürnberger
[1] Otto-von-Guericke University of Magdeburg
School of Computer Science
Universitätsplatz 2, 39106 Magdeburg, Germany
{deluca,nuernb}@iws.cs.uni-magdeburg.de

Abstract. In this paper we present an information retrieval system (meta search engine) that provides web service based methods for user and query specific annotation of search results. The system currently supports two types of annotations: Categorization based on information extracted from user bookmarks (Intelligent Bookmarks) and semantic disambiguation of query terms in documents using an ontology that can be selected by the user (Sense Folders). We describe both approaches and present adaptive user interfaces for different devices (mobile and desktop) that use these services in order to improve the user's search process. Furthermore, we discuss the advantages of dividing query results set processing (the information to be presented) from the interface design (information presentation) using web services in order to simplify the development of retrieval systems for, e.g., different desktop as well as mobile devices.

1 Introduction

Over the last years a number of methods have been proposed and realized, that enable a user to search in large collections of documents which may consist, e.g., of web pages or text documents. Most of these methods are based on keyword queries which means that a user has to provide a list of keywords that describe the contents of the searched document. After performing the query the user obtains a list of documents, which is ordered by the degree of similarity to the applied query (see, e.g. search engines like AltaVista [10] and Google [4]). If the keywords are well chosen, these methods frequently provide an appropriate list of results due to their sophisticated ranking methods, which are usually not based on pure Boolean queries but take into account word frequencies, thesauri or even simple semantics [1]. However, if the result list covers, e.g., diverse meaning categories (if the search terms are ambiguous) or diverse topic categories (if the search terms are used in different domains), then these categories appear rather unsorted in the result list. Since this is the case for most queries, automatic categorization of the documents would strongly improve the retrieval performance for a user, since he can then select the intended category and thus reduce the result list to a subset of relevant documents.

First approaches based on this idea had been provided, e.g., by the web search engine Yahoo [24]. Here, the complete document collection was (manually) hierarchically organized and this structure was provided to the user to browse the document collection. The main drawbacks of this approach are the necessity of manual interaction for the arrangement of the documents. As the classification has to be updated regularly, maintenance can get rather costly. More recent approaches which try to categorize documents automatically are realized, e.g., by meta search engines like Vivisimo [22] or KartOO [14]. Vivísimo's search and clustering solutions are based on an approach that organizes search results into categories. There is no preexisting taxonomy. The document clustering and meta-search software automatically categorize search results on-the-fly into hierarchical clusters. The categories that are automatically selected from the words and phrases contained in the results or documents themselves. The Vivísimo algorithm is based on textual similarity. Kartoo is a search engine which shows results in visual interface. The results of several search engines are combined and represented in a series of interactive two-dimensional maps through a proprietary algorithm. The thematic relations between the results are indicated with annotated lines. Furthermore, colors are used to symbolize certain subject. Thus, this search engine provides information in context in order to allow the user to navigate in semantic graphs.

However, all currently available categorization techniques still have difficulties in providing appropriate categories. Both, the manually assigned categories (subjectively labeled from humans) or the automatically derived categories (usually obtained by clustering methods [20, 3]) only consider the word distribution in documents without taking into account criteria derived from the underlying query, such as different meanings of a term or information from a user profile. Thus, the query as well as user specific interests are neglected during categorization. Therefore the assigned categories usually do not represent the categories a user is expecting for the query at hand.

In the following, we present the framework for an information retrieval system that enables the use of user profiles in order to automatically annotate result set based on user and query specific information. Furthermore, the system enables the use of user specified ontologies for disambiguation purposes and supports the use of adaptive interfaces on different devices (Sect. 2). Before we present in Sect. 4 two adaptive interfaces, we discuss in Sect. 3 the role of content and user adaptivity in information retrieval in order to motivate the specific interface design. In Sect. 4 we finally present two application examples and clarify the importance of the differentiation between content and presentation.

2 The Information Retrieval Framework

The main objective of user modeling in the area of information retrieval is to extract and store information about a user in order to improve the retrieval performance. A user model in this context usually consists of a list of keywords to which relevance degrees are assigned. More complex models [25] distinguish between different search contexts, or store additionally relations between keywords in order to model a more expressive search context. A user profile, based on the user interests, can be obtained, by extracting keywords from the queries performed or documents read by the user in the past [22]. Furthermore explicit as well as implicit feedback information from a user can be collected in order to learn or refine a user profile [17, 18, 2].

In the system presented in the following, we use user and query specific information in order to annotate – and thus categorize – search results from other search engines or text archives connected to the meta search engine by web services (see Fig. 1). The system currently supports two types of annotations: Categorization based on information from user bookmarks (*Intelligent Bookmarks*) and semantic disambiguation of query terms using an ontology that can be selected by the user (*Sense Folders*).

The idea of *Intelligent Bookmark* annotation is to exploit information about the way a user is ordering, sorting or categorizing his documents in order to categorize so far unseen documents. This approach is described in more detail in Sect. 2.2.

The idea of the *Sense Folder* annotation approach is to use ontologies in order to disambiguate query terms used in the retrieved documents [8]. Thus it is possible to categorize documents with respect to the meaning of a search term. Further details can be found in Sect. 2.3.

In the following we briefly describe the query processing and annotation process of the system and its embedding in the meta searcher.

2.1 Annotating Result Sets

In Fig. 1 an overview of the system architecture is given. The search engines (e.g. Google or local searchers) as well as the user interface are connected to the system by Web Services. Thus the system can easily be extended by additional search engines or used by different interfaces. The annotation methods are implemented as modules within the meta search engine.

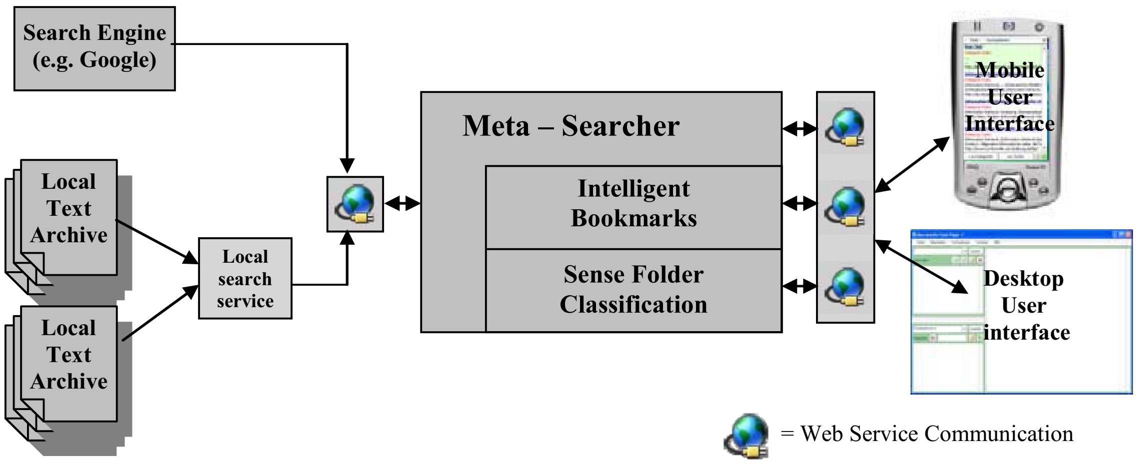

Fig. 1 Overview of the retrieval system

The main idea of the system is to provide additional disambiguating information to the documents of a result set retrieved from a search engine in order to enable the clients to annotate, restructure or filter the retrieved document result set. This idea is motivated by user studies have shown that category interfaces are more effective than list interfaces in presenting and browsing information. For example, in the studies presented in [11] the authors have evaluated the effectiveness of different interfaces for organizing search results. It turned out that users favored the category interface over the list interface and were 50% faster in finding information organized into categories. In this study the authors provided additional semantic category contexts for the list interface using categories as described in [15] in order to also evaluate what interface elements were the most important for the search process.

The annotation process we have implemented into the meta search engine can be briefly described as follows:

1. First the user types his query (keywords).
2. Search results are indexed.
3. Search results are classified/annotated by its *Sense Folder*.
4. Search results are classified/annotated using the *Intelligent Bookmarks.*

Based on this process an annotated result set is obtained. This result set is forwarded to the client via the web services.

2.2 The Intelligent Bookmark Approach

Most users use the bookmark functionality of their web browser to store relevant websites in a more or less structured way. The main idea of the *Intelligent Bookmarks* is to provide users additional benefit from storing bookmarks. Therefore we extended the functionality in different ways: The system supports hierarchical bookmarks in a tree structure, the bookmark hierarchy can be accessed and managed from any device and furthermore the structural information stored by the bookmark hierarchy and the assigned web pages is used to annotate search results. All functions can be accessed via web services. Thus, the bookmarks can be visualized by clients in different ways, e.g., as a folder structure like in the interfaces discussed in Sect. 4.

Based on the web pages stored in the bookmark structure, a classifier is trained that uses the folder names as category labels. Thus, the more results are stored and assigned to a category, the better does the system learn something about the way a user is structuring information. We assume that every category folders describe groups of web sites that implicitly define the categories for the support system of the search interface.

This classification approach can be used in order to annotate search results or to filter documents. Since the system just provides the annotation, the visualization for the user can be realized by the client systems. A more detailed description of the functionality of this process is given in [13].

2.3 The Sense Folder Approach

Similar to the *Intelligent Bookmark* annotation, the classification terms obtained by the disambiguating classes of the *Sense Folders* are then added to each document listed in the result set of a query and are forwarded to the search client for further use. In the *Sense Folder* approach integrated in the retrieval system we consider the different linguistic relations that describe the context of the searched word in the ontology in order to recognize the meaning of the user query [8]. Currently we use these linguistic relations from the WordNet ontology [23] in order to create prototype vectors that are defining the *Sense Folders* for the different meanings of the query terms. Obviously, this approach is restricted to query terms that appear in the ontology.

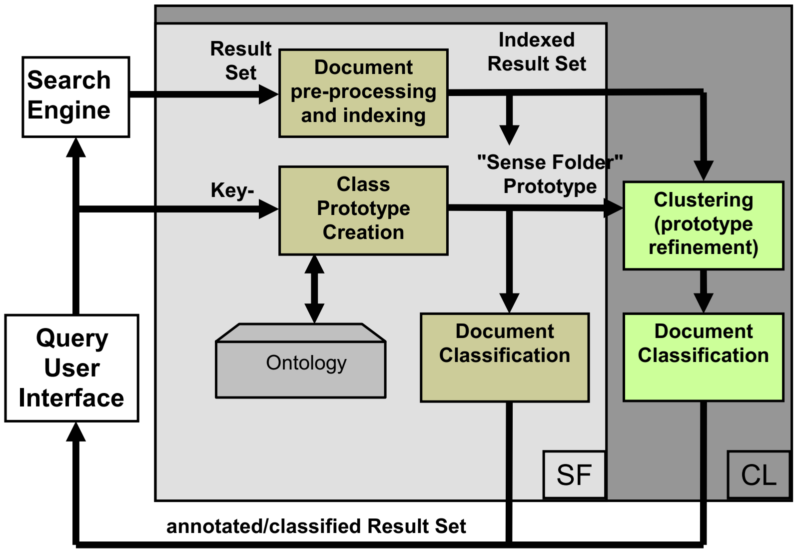

Fig. 2 Overview of the Sense Folder (SF) and Clustering (CL) classification process

For our disambiguation problem, we first assign semantic information (retrieved from the ontology) to the result sets we get from the search engine. Afterwards, we use a clustering algorithm in order to fine tune the initial prototype vectors of each *Sense Folder* using the distribution of documents around the initial prototype vectors (see Fig. 2), i.e., we expect that in a web search usually a subset of documents for each possible meaning of a search term is retrieved. Thus, each subset forms a cluster in document space describing one semantic meaning of this term. In other words, every document is first assigned to its nearest prototype vector derived from the ontology and afterwards this classification is revised by the clustering process [9]. This approach has shown to strongly improve the classification (or disambiguation) performance [9]. The semantic information assigned is appended to the document as additional information in order to help the user in finding the relevant documents, without loosing too much time in browsing all documents.

3 User and Content Adaptivity for Information Presentation

Standard keyword based search engines retrieve documents without considering the importance of user oriented information presentation. It means that the user has to analyze every document and decide himself which are the documents that are relevant. In order to support the user the retrieval system assigns additional information – currently retrieved from the ontology or the bookmark structure as described above – to the retrieved documents. Furthermore, this information should also be adapted according to the individual user needs.

The need to obtain a good adaptivity [5, 6, 7] to user needs should be based, on an appropriate user model in the retrieval system, since users expect individual information depending on their interests and knowledge. In order to achieve this, we need different user profiles [19, 21] that cover (almost) all user needs, including the relations between the user search words (content adaptivity), and the different user characteristics (depending on the user interests, knowledge, experience, language, culture, etc.). In this sense the *Intelligent Bookmarks* as described in Sect. 2.2 are only a first step. However, since the system exploits information the user is anyway willing to provide, this approach is non-obtrusive and thus more easily accepted then explicit feedback techniques. By storing bookmarks the user implicitly provides information about his interests and the way he likes to have information structured.

Besides the information describing a user we have to consider certain limits of the hardware when we implement a user interface for a specific, e.g., mobile device, and have to adapt our interface accordingly. We also have to observe, that information has to be independent from the interface and has to be presented individually; it means that users should have a unique support depending on their needs as independent as possible from the hardware they are currently using.

Implementing our retrieval system, we decided to divide query results set processing (the information to be presented) from the interface design (information presentation) using web services in order to simplify the development of retrieval systems for, e.g., different desktop as well as mobile devices. We have chosen to implement web services in order to give the possibility to access information in a way that is platform and hardware independent. Thus the communication between the search-client and the meta search engine as well as the meta search engine and search engines like Google or local search services is realized trough web services that represent a standardized interface. For the current implementation of the meta search engine we used Axis – an open source SOAP server and client – on a Tomcat Web Server. As an example, in Figure 3 a subset of the web services that are necessary for accessing the basic search functionality and the bookmark management is shown.

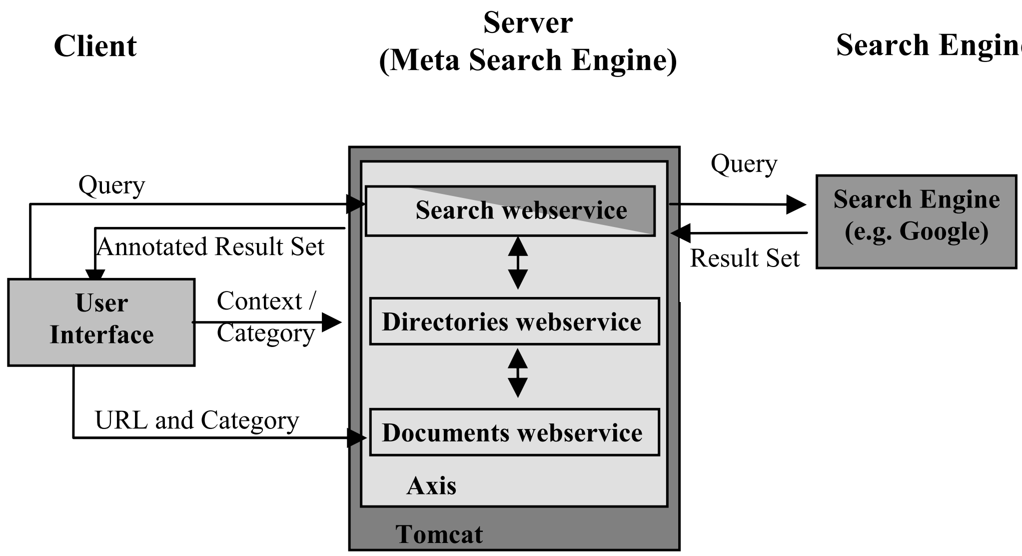

Fig. 3 Web Services interaction overview using the bookmark based classification

The first web service (*Search*) connects to a search engine (here Google using the Google API) and returns a set of results that contains additional annotations as described in Sect. 2. The *Directories* web service is used to manage the bookmark categories created from the user. Therefore it provides methods to add and remove folders for the bookmark hierarchy. The *Documents* web service is used to store and remove links (URLs) to web pages in the bookmark hierarchy.

4 User Interface Design

Since an information retrieval interface that is developed based on the meta search engine discussed above should help to accelerate the search process of a user, it is important to design the interface as user friendly as possible. Therefore in this section we discuss two prototypical search interfaces we implemented for the visualization of the available data. We also discuss briefly the different requirements that have to be considered implementing user interfaces for diverse devices.

4.1 Desktop User Interface

A screenshot of a prototypical search interface for desktop systems is shown in Figure 4. The interface basically consists of three parts: The query area, the contexts and categories area, and the result list area.

The query area is in the upper left side of the interface. Here, the user can enter the search terms that will be used from the system in order to obtain a list of results from a connected search engine. The obtained results are then stored by the interface in a specific folder that is labeled with the search terms given by the user. Every time a user starts another query, a new "query" folder will be created, thus the user can navigate through them without writing the same query twice.

The bookmark area is placed underneath the query area. The bookmarks are user specific. Here we used the first level in the bookmark hierarchy to distinguish between search context and categories within a given search context. Thus the user can store bookmark hierarchies for different search purposes, e.g. work and recreational activities. This supports on the one hand the user in structuring the bookmarks and on the other hand simplifies the classification of documents in search results, since we already have separated the documents in different domains and can learn individual classifiers for the respective subsets [13].

The result list area is situated on the right side. Here both, the web search results and the list of web sites contained in a selected category, are presented. For each item its title, its hyperlink, and a snippet briefly describing the content of the belonging web page as provided by standard search engines is given. In addition, categories derived by the *Intelligent Bookmark* and *Sense Folder* annotation methods (see Sect. 2) are displayed. For the *Sense Folder* annotation additional label information is retrieved from the ontology and displayed to give the user a more detailed description of the content of the annotated document entry. Thus documents belonging to different semantic categories are labeled with different senses derived from the ontology (see also Section 2.3 and [8]).

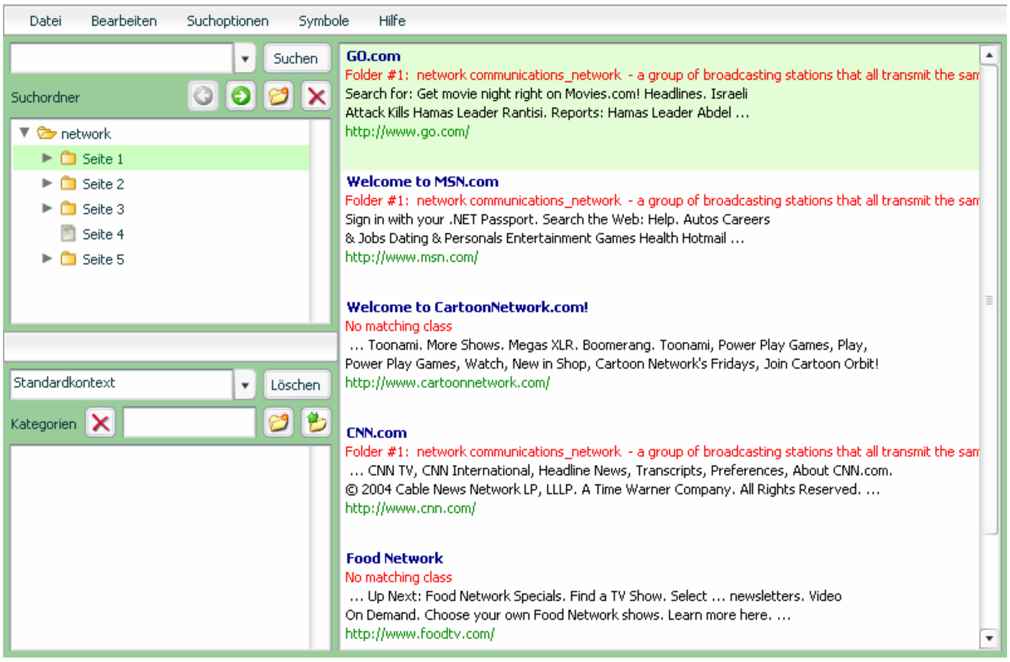

Fig. 4 Desktop User Interface. The information from the Sense Folder annotation is provided directly below each title line.

4.2 Mobile User Interface

The first consideration has to be done working on mobile devices is why such an effort is so different from the interface design for a more typical desktop based Web application. In the following, we mention some problems faced by user interface designers in wireless application development: Typically a wireless device has a more limited bandwidth than a wired device. Transmitting and receiving huge amounts of data is a problem. Therefore, the data containing the information to be presented should be well preprocessed and redundant, and unnecessary data should be removed.

Furthermore, the connection to a wireless device is intermittent and there is no persistent point-to-point connection. Mobile devices are compact in size and have the problem of the limited battery life. We have also the problem of limited memory – that, however, sometimes can be resolved expanding it with memory cards. Another fundamental difference between mobile devices and standard workstations is the user interface. A “normal” interaction (it means through mouse and/or keyboard) for such devices is usually not available. Furthermore, the screen area is almost always very small and thus data can be viewed, navigated and manipulated usually only in a very cumbersome way if the user interface has not been adapted for this environment [12, 16].

Using mobile devices, we had to consider the different possibilities of user interaction. We transferred the given functionality provided by the desktop interface, as good as possible to the PocketPC. The realized interface is shown in Figure 5a-c.

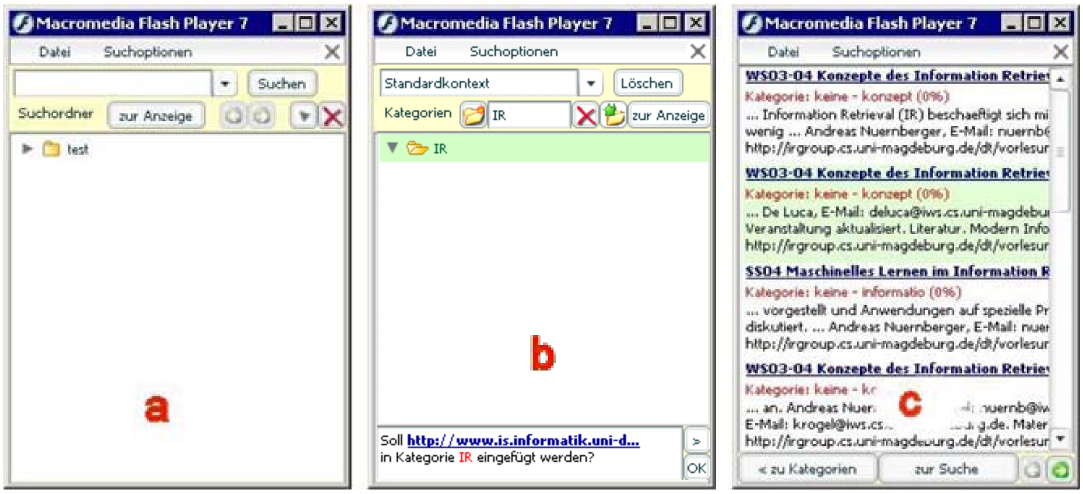

Fig. 5 Mobile User Interface: a) Search window, b) bookmark management and c) result list

In order to ensure an intuitive use, we provide three different adapted views of the user interface. In the desktop user interface, for example, we have a drag and drop functionality, in order to store or restructure the bookmarks. This is not possible in the mobile environment, because of the limited interaction functionality given from the use of the pen. For this reason we had to readapt the functionality of the user interface.

The navigation through the result list on a mobile device is done with a pen. If the user clicks on a document, the content will be then shown as a normal web page in the window of the standard web browser of the device.

Taking into account all software and hardware limitations, we developed an adapted mobile user interface. The basic components are three. We adapted their functionalities for mobile devices. These components (views) of the program are listed as it follows:

a) Search and presentation of the search results in a tree-form.
b) Results window: Search results are annotated/categorized using the different classification approaches (see Sect. 3) and are presented with the additional information as first information after the title.
c) The third component is used for the user specific (private) categories and contexts (bookmarks).

The user can type the query in the search window (Fig. 5a) using standard PDA text input methods. The system retrieves the documents presenting automatically the results, switching to the results window (Fig. 5c). Once a user gets results, he can choose to see one document by clicking on it with the pen. He can view the next results choosing the "next arrow" or can start a new search clicking on the search button.

Buttons (Fig. 5) provide the possibility to switch between the views. In every view a user can switch to another only clicking to the correspondent button. The user can choose to search new content, browsing the results, saving the interesting results to the *Intelligent Bookmarks* or viewing new results trough the simple use of a pen. All these interaction possibilities are given by the use of the buttons that give a quick access to the other view. Storing data in categories and context is possible by selecting a document and pressing the “category button”. This implements the drag and drop functionality of the desktop interface (see also Figure 5b).

We implemented the user interface using the Macromedia Standalone-Flash Player for PocketPCs. Therefore, the user interface is portable to any device that has installed the player. If a Macromedia plug-in for the Pocket Internet Explorer is installed, the user interface can also be accessed as macromedia flash-film. However, since the interface is then re-scaled this has negative effects on picture quality, performance and usability. The PDA connects to the web services provided on our web server using an internet connection, e.g., via WLAN, Bluetooth or USB-connection (ActiveSync).

5 Conclusions

In this paper we have presented the concept of a meta search engine that can be used by desktop as well as mobile information retrieval interfaces. The system supports two types of annotations: Categorization based on information from user bookmarks (*Intelligent Bookmarks*) and semantic disambiguation of query terms using an ontology that can be selected by the user (*Sense Folders*). We have briefly discussed these approaches and how to implement them for building adaptive user interfaces for different devices (mobile and desktop) in order to accelerate the user’s search process. Furthermore, we discussed the advantages of dividing query results set processing (the information or content to be presented) from the interface design (information presentation) using web services in order to simplify the development of retrieval systems for standard desktop workstations as well as mobile devices.

References

1. Agosti, M., Crestani, F., and Pasi, G., Lectures on Information Retrieval, Lecture Notes in Computer Science. Vol. 1980, Springer Verlag, 2001.
2. Bade, K., De Luca, Ernesto W., and Nürnberger, A. Multimedia Retrieval: Fundamental Techniques and Principles of Adaptivity, In: Künstliche Intelligenz, 04, 2004.
3. Borgelt, C., Nürnberger, A., Fast Fuzzy Clustering of Web Page Collections, In: Proc. of PKDD Workshop on Statistical Approaches for Web Mining (SAWM), Pisa, Italy, 2004.
4. Brin, S., and Page, L. , The Anatomy of a Large-Scale Hypertextual Web Search Engine, In: Proc. of the 7th Int. World Wide Web Conf., pp. 107-117, Brisbane, Australia, 1998.
5. Brusilovsky, P., Methods and techniques of adaptive hypermedia, in User Modeling and User Adapted Interaction, 6:2-3, 1996.
6. Brusilovsky, P., Kobsa, A., Vassileva, J., (eds.), Adaptive Hypertext and Hypermedia, Kluwer Academic Publishers, 1998.
7. Brusilovsky, P., Stock, O., Strapparava C., (eds.), Adaptive Hypermedia and Adaptive Web-Based Systems, Proc. of the International Conference AH 2000, Trento, Italy, 2000.
8. De Luca, E. W., Nürnberger, A., Ontology-Based Semantic Online Classification of Documents: Supporting Users in Searching the Web, In: Proc. of the European Symposium on Intelligent Technologies (EUNITE 2004), Aachen, 2004.

9. De Luca, E. W., Nürnberger, A., Improving Ontology-Based Sense Folder Classification of Document Collections with Clustering Methods, In: Proc. of the 2nd Int. Workshop on Adaptive Multimedia Retrieval (AMR 2004), pp 72-86, Valencia, 2004.
10. Digital Equipment Corporation, AltaVista, http: //www.altavista.com/, 1995.
11. Dumais, S. T. Cutrell, E., and Chen, H., Bringing order to the web: Optimizing search by showing results in context. In Proceedings of CHI'01, Human Factors in Computing Systems, pp. 277-283, 2001.
12. Eisenstein, J., Vanderdonckt, J. and Puerta A., Adapting to Mobile Contexts with User-Interface Modeling, In: Proc. of the Third IEEE Workshop on Mobile Computing Systems and Applications (WMCSA'00), pp. 83-94, 2000.
13. Grabski, K., Nürnberger A., A Search Interface for Context Based Categorization of Result Sets, In: Proc. of the European Symp. on Intel. Techn. (EUNITE 2004), Aachen, 2004.
14. Kartoo Homepage, http://www.kartoo.com/.
15. Labrou, Y. and Finin, T., Yahoo! as an ontology: using Yahoo! categories to describe documents. Proceedings of the eighth international conference on Information and Knowledge Management, Kansas City, Missouri, 1999.
16. Lam, H. and Baudisch, P., Summary Thumbnails: Readable Overviews for Small Screen Web Browsers, in Proc. of CHI 2005, Portland, OR, pp. 681-690, 2005.
17. Rocchio, J.J., 1971, "Relevance Feedback in Information Retrieval", IN: Salton, G., "The SMART Retrieval System", pp. 313-323, Prentice Hall, Englewood Cliffs, NJ.
18. Ruvini, J.-D., Adapting to the User's Internet Search Strategy. In Proceedings of the 9th International Conference on User Modeling (UM2003), Pittsburgh, 2003.
19. Stefani A., Strapparava. C., Exploiting NLP techniques to build user model for Web sites: the use of WordNet in SiteIF Project. In P. Brusilovsky, P. De Bra & A. Kobsa (Chairs), Proc. of the 2nd Workshop on Adaptive Systems and User Modeling on the WWW, 1999.
20. Steinbach, M., Karypis, G., and Kumara, V.. A Comparison of Document Clustering Techniques, In: KDD Workshop on Text Mining, 2000.
21. Strapparava, C., Magnini, B., and Stefani A., "Word Sense Based User Models for Web Sites". 9th Int. World Wide Web, The Web: The Next Generation. Amsterdam, 2000.
22. Vivísimo Homepage, http://vivisimo.com/.
23. WordNet homepage, http://www.cogsci.princeton.edu/~wn/, („Five Papers on WordNet"), 2004.
24. Yang, J., and Filo, D. (1994). Yahoo Home Page, URL: http://www.yahoo.com.
25. Zhu, T.,Greiner, R.and Haubl, G.. Learning a model of a web user's interests. In The 9th International Conference on User Modeling (UM2003), Pittsburgh, 2003.

Adapting Information Delivery to Groups of People

Judy Kay and William Niu

School of Information Technologies
University of Sydney, Australia
{judy, niu}@it.usyd.edu.au

Abstract. When people engage with information, they are often in social groups. This applies, for example, in the case of museum visits, where people typically attend the museum and view the exhibits in small groups. This paper describes a proposed system GM, Group Modeller, which will create group models from a set of individual user models. We discuss the approaches and challenges in terms of common sub-models, collective models, group interaction models, and knowledge-based reasoning across models.

1 Introduction

With computing technology becoming pervasive, information access will become increasingly integrated into normal environments. For personalisation researchers, it is important to begin to take account of the many everyday situations where the user of a personalised service is not alone, but is part of a small group. If we are to customise or adapt information delivery, we can expect that this will operate differently depending on whether a person is alone or in a group. A suitable individual user model may suffice in the situation when a person is alone. It is conceivable that a heterogeneous user group may have conflicting preferences and needs, and the delivery of customised information which addresses the requirements of all the people in the group introduces new challenges in addition to those relevant in the case of an individual. When information is to be adaptive in a group setting, the system will need a group model by appropriately amalgamating the individual models.

There has been some interesting and relevant work on user modelling in the context of personalised museum tours (many of which are reviewed in [1]). As one might expect in this context, some of the work takes account of groups in their design (e.g. [2], [3], and [4]). Petrelli, Angeli, and Convertino [5] observed that people exhibit different behaviour when visiting a museum in a group, as opposed to visiting as an individual. Moreover, a person's behaviour typically depends on the members that make up the group; for example, a young child will typically play a predominant role in determining the duration and the course of a family visit in a museum, with adults accommodating the child's needs and preferences rather than their own.

Another domain that we want to apply the Group Modeller (GM) in is the domestic environment, which differs from museum settings in several aspects (e.g. group composition, session duration, etc.). Unlike a museum, the group members in a domestic environment are usually quite tightly knit groups like families. These people typically have some traits in common (e.g. social values and habits). In addition, and more importantly from a personalisation perspective, since the home is where many people spend large periods of their time, it seems likely that it may be easier to maintain long-term, and perhaps more accurate, user models for people in that context. We are also interested in making use of user models built and maintained in a domestic environment, and then use them in other environments, such as museums.

In the Section 2, we discuss work on group modelling in different domains. Section 3 examines some challenges in composing an effective group model. Next we will describe the proposed group modelling system GM, followed by the conclusions.

2 Group Modelling

Much work has been done in addressing groups in various domains. Masthoff [6] investigated how different group decision rules affect the order of a sequence of numerically rated preferences (in this case, for TV programs) and the satisfaction gained as a whole group by applying each rule. She conducted two experiments. One was, from a third-person view, on how people select a series of TV programs for a group of viewers. In the second one, she examined how satisfied people would feel with the sequences produced by the different strategies. Some interesting findings were that: people tried to account for fairness and avoid individual misery; normalisation was used (i.e. their satisfaction is based on both selected and non-selected items); ratings were judged in a non-linear way (e.g. in a 10-point scale, the difference between 9 and 10 is more significant than that between 6 and 7). This type of work has importance beyond the choice of TV programs. It suggests strategies for dealing with conflicting preferences in more general contexts.

MusicFX [7] is another example of a system that uses individual user models to generate group models. MusicFX is used in a fitness centre to adjust the selection of background music to best suit the preferences of the people exercising at any given time. One interesting facet of the system is that a group in the context is made up of the people who happen to work out at the same time. This is a very different sense of a group from that in most other projects (e.g. [2], [6], [8]), where the group is not composed of strangers; rather it is friends or family members. This system is somewhat unusual in that it uses *explicit* preferences of all participants to make a selection that will directly affect everyone who is present.

Group modelling is also an important factor in some work on city tours. In the INTRIGUE project, for example, attractions are separately ranked by first partitioning a user group into a number of *homogeneous subgroups* with the same characteristics. Then each subgroup may fit one or more stereotypes. Finally, the subgroups are combined to obtain the overall preference, in terms of which attractions to see, for the whole group [9]. In some cases, a subgroup could be particularly influential either because it contains a majority of members of the group or because it represents a relevant tourist class (e.g. children and disabled people). For example, a subgroup may have the following characters: between age of 46 to 55, full mobility, partial vision, and interested in art. One of the stereotypes the subgroup fits is vision-impaired, which would put a high weighting on choosing attractions that facilitate people with impaired vision (e.g. vocal presentation and/or visual aids).

Another well-known group recommender system is the Travel Decision Forum prototype that supports a group of people to plan joint vacations [10]. Inspired by situations where face-to-face communication is not possible, this system emphasises asynchronous group discussions. In the initial phase, each group member specifies his or her preferences by filling in a *preference specification form*. The aim of the next phase is to reach uniformed agreement by having each group member interacting with a virtual mediator, as well as the virtual agents that represent other members. This system introduces a novel type of *incremental preference elicitation*; as each member fills up a preference specification form, he or she may choose to see example solutions based on the preferences of all the group members. Another interesting aspect is how to minimise *manipulative* preference specification. For example, when a person sees the overall rating of an activity is fairly positive, he might rate it lower than what he would have without seeing others' ratings, in order to leverage the final outcome. This still remains a challenging issue involving difficult tradeoffs.

There is rather scant literature on group modelling in the home and museum settings. Sotto Voce [2] was designed to accommodate a group of users in a museum tour, but did not include adaptation as a requirement. Kay, Lum, and Niu [4] presented a scenario on how a scrutably adaptive museum guide may deliver personalised information to each pupil in a school group, which in turn stimulates after-visit group discussions. PEACH is perhaps one of the most ambitious projects in museum research. They discuss issues of how to adapt the information to a small user group. For example, Kruppa [8] explores aspects of providing some common information to the group on a large display and some personalised information on a hand-held display.

Adaptation in homes has involved customisation to the inhabitants as a whole. Voida and Mynatt's [11] experiment on probing families' values reveals a possible approach to designing an information adaptive environment. On the other end of scale, the Casablanca project has designed several prototypes for the home, stressing social communications between family members [12]. Although the devices facilitate communication within the family, it does not provide adaptation for each family member.

3 Challenges in Group Modelling

We describe three of the basic approaches to combining individual user models. Then we describe approaches to enhance reasoning about a collection of individual models in the very likely case that different aspects of people are modelled in the different user models. We need to reconcile these differences to make a more complete and effective group model. We begin by introducing an example. Table 1 lists four hypothetical people's individual user models of preferences on entertainment, which will be referred in the rest of the section. A tick ($\surd$) indicates a positive preference, an X means a negative preference, and a blank space represents an unknown preference. So, for example UM_A models person A, who likes Horror movies, Documentaries, and Musicals, but it does not model the person's preferences for Cartoons or Jazz.

	UM_A	UM_B	UM_C	UM_D
Horror movies	$\surd$		$\surd$	
Documentaries	$\surd$			
Musicals	$\surd$	$\surd$	X	
Cartoons		$\surd$		
Jazz				$\surd$

Table 1. Exemplary Individual Models

Common Sub-models

This is probably the simplest way of combining individual models; it simply involves grouping the properties (or preferences) that all individual models share. We call this a **common sub-model**. For example, when two people, represented by UM_A and UM_B in Table 1 respectively, would like to find a show or movie to watch, this approach would suggest a musical. This approach is essentially a logical AND over the individual models to create the group model.

Collective Models

Unfortunately, a common sub-model often may not provide enough information to compose an adequate group model. In this case, some collective properties have to be chosen to complement the common sub-model. An obvious approach is to perform an operation like a logical OR on the set of individual models to form the group model. We call this process **collective modelling**. Sensibly, the properties that are of the interest to a majority of the group are preferred.

Now take UM_A, UM_B, and UM_C from Table 1 as an example. While there is no one common preference between the three subjects, horror movies and musicals both have two votes. The person with UM_C, however, explicitly expresses a negative preference against Musicals. As a result, a movie with a horror theme may be a more satisfying choice for this group.

Group Interaction Models

The two above approaches are extremely simple-minded. They totally neglect the fact that people react to others in the group. An alternative approach to collective modelling is to account for group composition and social interaction within the group members, which we name **group interaction modelling**. This opens up a wide variety of possibilities and has been a popular research field in group modelling in the past few years. For example, a few group decision strategies discussed in [6], such as the Average Without Misery Strategy, the Fairness Strategy, and the Dictatorship, address group interaction between a group of TV viewers. Those strategies require the group members to have numerical ratings for each property, which represents a TV program in this case.

Another interesting approach that uses the group interaction modelling from a different perspective is the INTRIGUE project [9] reviewed in Section 2. It proposed to partition a user group into a number of *homogeneous subgroups*. Each subgroup had different influential power on the decision making process, and the power might be caused by the size of the subgroup and the class of the subgroup members.

Again, use UM_A, UM_B, and UM_C from Table 1 as an example. While a horror movie may be a best selection using collective modelling, the decision may well be altered if the person represented by UM_B is a 10-year-old child. Because of the adequacy problem, a musical may be chosen despite the subject with UM_C has a negative preference, or either a documentary or a cartoon movie may be chosen under some other group decision strategies.

Knowledge-based Reasoning

The above approaches rely upon the existence of common components in the user models; expressed differently, this means they require that the different user models have a common vocabulary. Where this is not the case, we need inference mechanisms to overcome the problem. This section briefly outlines some of the important forms that this will take.

Figure 1 shows an example of two individual user models. Each circle denotes the user model namespace; it represents all the components modelled. Some of these may not yet be known. For example, UM_1 models user preferences for Tea, Cheese, and Jam, but not Wine. The intersection of the two circles represents the group model between the two individual models. A black dot is a positive preference of a property (e.g. Tea in UM_1). A white dot is a negative preference of a property (e.g. Jam in UM_2). In the case where a component is at one user model but not the other (e.g. Wine in UM_2), it means the component is only modelled in that user model.

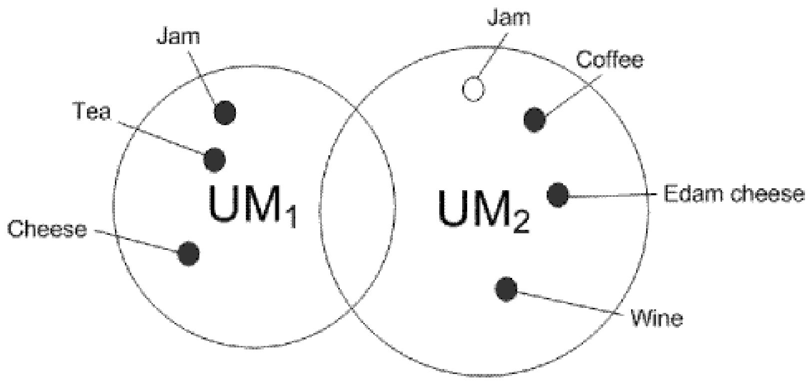

Fig. 1. Exemplary User Models

Common sense reasoning– This is an important part of making sensible assessments about people. We have already alluded to one example of this, in the case of a group which includes adults and children choosing a TV show. It is common sense that the adults will ensure that the children's needs are given priority and adults would expect to watch child-appropriate programs.

Figure 2 gives an example of how the graph would look after applying common sense reasoning to Figure 1. If person 1, represented by UM_1, is a child, he or she normally is not allowed to drink wine, regardless whether he or she likes it or not. So the preference for wine is negative when those two people spend time together.

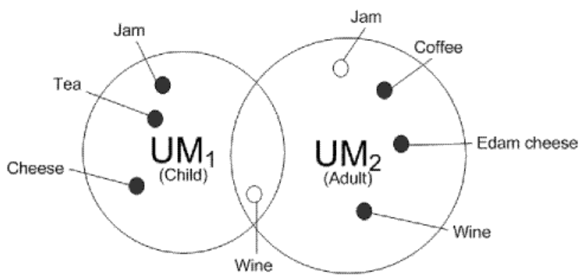

Fig. 2. Applying Common Sense Reasoning

Stereotypic reasoning – Stereotypic reasoning is an important form of user modelling inference that uses statistically valid generalisations to quickly start up a user model [13]. It uses a *trigger*, in the form of simple, readily available information to make a large number of low quality default inferences; they should be overridden once more reliable evidence is available. To give an example of stereotype, if a person is known to be a university professor, it may be suggested that he or she is intellectual, well-educated, fairly wealthy, honest, male, over forty, and well travelled.

This reasoning may also be applied on groups to explore more similarities among a group of people. Say, a group of three people, represented by UM_A, UM_B, and UM_D from Table 1, would like to settle on something to do during a weekend. Note that there is little information about the person with model UM_D. Suppose, we have stereotypic knowledge that people who like jazz typically like musicals and horror movies. In this case, a musical may be recommended as the social activity between the three people.

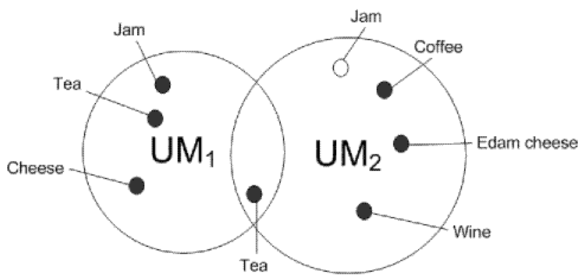

Fig. 3. Applying Stereotypic Reasoning

As a graphical example, Figure 3 illustrates how the example shown in Figure 1 would be after applying stereotypic reasoning. Suppose one of the stereotypes infers that people who like coffee also like tea. This stereotype infers that person 2, who likes coffee, also likes tea. Hence the positive preference of tea for the group model.

Ontological reasoning – Another important form of reasoning for user modelling is ontologically-based. This is important for determining relationships between vocabularies, hence establishing connections across the user models of different individuals. Figure 4 exemplifies the use of an ontology to reason the user models in Figure 1. Person 1 likes all cheese, and person 2 likes the Edam cheese. Hence it is likely that person 1 will like Edam cheese.

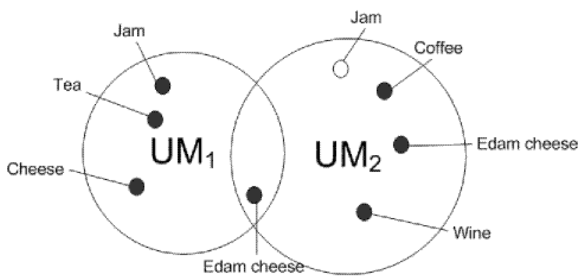

Fig. 4. Applying Ontological Reasoning

4 Proposed System Architecture

Figure 5 illustrates the envisioned architecture for the Group Modeller. We now describe the approaches and tools we propose to use in order to realise it.

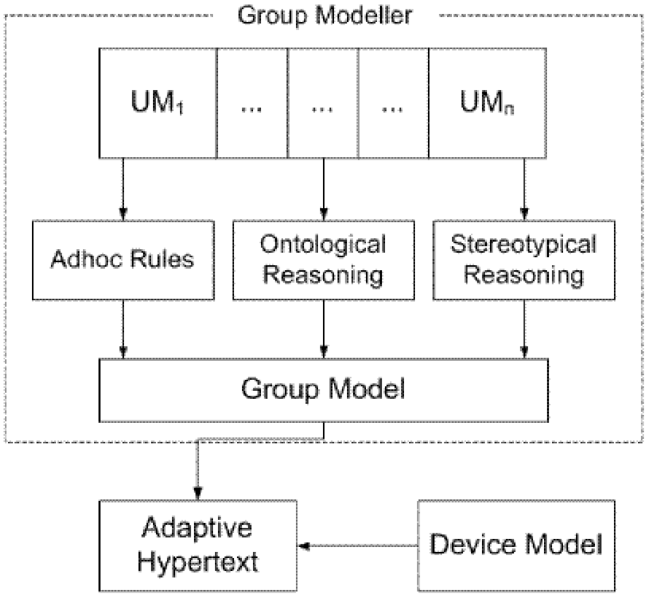

Fig. 5. Envisioned System Architecture

User Modelling Server

The personalisation of information delivery is powered by the user model. Actions by the user are stored in the user model as evidence. The evidence may range from the duration of time the user has spent viewing or interacting with a particular museum exhibit to the history of exhibits he or she has visited. Based on this evidence the system can draw conclusions about user preferences and from this, tailor the delivery of information.

The user modelling server Personis [14] allows adaptive systems to easily manage evidence for user models, and provide a resolution system to conclude a value of each user model component based on this evidence. These resolvers are crafted by the system designers with scrutability in mind. At any time, users should be able to ask the system why an adaptation was performed, and the system should respond with the evidence that lead to the adaptation. With this in mind, the same resolvers can be accessed by different devices, with the results tailored at the device level to be appropriate to the interface. These properties also make Personis a suitable candidate for modelling groups. We simply need to establish suitable approaches for the new resolvers that will reason about groups, rather than just individuals.

Ontological Reasoning

We propose to model this knowledge with a light-weight ontology. MECUREO [15] is one such tool to fit this task. It was originally designed to create an ontology of computer science terms from the Free On-Line Dictionary Of Computing (FOLDOC) and has since been used in several experimental systems, using various dictionaries/glossaries.

Stereotypic Reasoning

In the Personis approach, this is managed by a knowledge source that provides evidence. That evidence is distinguished as stereotypic, and resolvers treat this as less reliable than other forms of evidence. We envisage that the Group Modeller would need to establish which components in the various individual user models need additional evidence. Backward chaining through stereotypes could be used to search for suitable stereotypes to support reasoning about these for the group.

Adaptive Hypertext

We have been developing a version of the Scrutable Adaptive Hypertext system [16] that integrates the Personis user modelling server. The web-based interface, adaptability, and controls for scrutability make it a suitable medium for the system described above. Each page is tailored to the user(s). Whole pages may be omitted. At any time the user may choose to see how the page currently viewed is adapted to her or him. The text being included or excluded is highlighted with different colours. By moving the mouse cursor over each section of the highlighted text, the reason for inclusion or exclusion is provided. A basic description of the user model is also displayed to the right of each page.

5 Conclusions

There are many situations where information should be delivered to a group of users. This requires that we manage group models. We have explored some of the issues involved in doing this, identified some approaches that should be part of a Group Modeller and presented the architecture of an experimental system that will be a testbed for group modelling.

Acknowledgements

This project is partially funded by Smart Internet Technology CRC.

References

1. Filippini-Fantoni, S.: Museums with a personal touch. In: EVA 2003 London Conference Proceedings. (2003)
2. Aoki, P.M., Grinter, R.E., Hurst, A., Szymanski, M.H., Thornton, J.D., Woodruff, A.: Sotto voce: Exploring the interplay of conversation and mobile audio spaces. In: CHI '02: Proceedings of the SIGCHI conference on Human factors in computing systems, Minneapolis, Minnesota, USA, ACM Press (2002)
3. Stock, O., Rocchi, C., Zancanaro, M., Kuflik, T.: Discussing groups in a mobile technology environment. In A. Buts, C. Kray, A.K.A.S.H.P., ed.: MU3I '05: 2nd Workshop on Multi-User and Ubiquitous User Interfaces 2005 in Conjunction with the IUI'05, San Diego, California, USA, ACM Press (2005)
4. Kay, J., Lum, A., Niu, W.: A scrutable museum tour guide system. In Butz, A., Kray, C., Kruger, A., eds.: MU3I'05: 2nd Workshop on Multi-User and Ubiquitous User Interfaces at IUI '05. (2005)
5. Petrelli, D., Angeli, A.D., Convertino, G.: A user centered approach to user modelling. In Kay, J., ed.: UM'99: Proceedings of the Seventh International Conference on User Modelling, Springer Wien New York (1999)
6. Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. User Modeling and User-Adapted Interaction **14** (2004)
7. McCarthy, J.F., Anagnost, T.D.: Musicfx: an arbiter of group preferences for computer supported collaborative workouts. In: CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work, Seattle, Washington, United States, ACM Press (1998)
8. Kruppa, M.: A user study on parallel presentations for heterogeneous user groups. Technical Report 563, Saarland University
9. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. Applied Artificial Intelligence **17** (2003)
10. Jameson, A.: More than the sum of its members: challenges for group recommender systems. In: AVI '04: Proceedings of the working conference on Advanced Visual Interfaces, Gallipoli, Italy, ACM Press (2004)
11. Voida, A., Mynatt, E.D.: Conveying user values between families and designers. In: CHI '05: Proceedings of the SIGCHI conference on Human factors in computing systems, Portland, Oregon, ACM
12. Hindus, D., Mainwaring, S.D., Leduc, N., Hagström, A.E., Bayley, O.: Casablanca: designing social communication devices for the home. In: CHI '01: Proceedings of the SIGCHI conference on Human factors in computing systems, ACM Press (2001)
13. Kay, J.: Lies, damned lies and stereostypes: pragmatic approximations of users. In Kobsa, A., Litman, D., eds.: UM '94: Proceedings of the Fourth International Conference on User Modeling. (1994)
14. Kay, J., Kummerfeld, B., Lauder, P.: Personis: A server for user models. In: AH'02: Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems, Springer-Verlag, London, UK (2002)
15. Apted, T., Kay, J.: Mecureo ontology and modelling tools. WBES of the International Journal of Continuing Engineering Education and Lifelong Learning **14** (2004)
16. Czarkowski, M., Kay, J.: A scrutable adaptive hypertext. In: AH '02: Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, Springer-Verlag (2002)

A Unified User Profile Framework for Query Disambiguation and Personalization

Georgia Koutrika, Yannis Ioannidis

University of Athens, Greece
{koutrika, yannis}@di.uoa.gr

Abstract. Personalization of keyword searches has attracted interest in the research community as a means to decrease search ambiguity and return results that are more bound to be interesting to a particular user. We describe a term-based user profile that treats query disambiguation and personalization as a uniform term rewriting process. Its key feature is the representation of connections between terms based on possible rewritings between them on a per user basis. We present a query-rewriting algorithm for such query disambiguation based on the proposed profiles. Preliminary experimental results show the potential of the overall approach.

1 Introduction

Traditionally, search engines are deterministic in that they should return the same set of documents to all users with the same query at a certain time. Therefore, it is inherent that search engines are not designed to adapt to personal preferences. This deterministic behavior is desired in order to provide the users with the same view of information; however, in the context of the World Wide Web, it often hinders users from locating relevant information. There are several aspects to the problem. First is the problem of abundant information made available to a wide spectrum of users with possibly different information needs. Only a fragment of this information is useful to a single user. Typically, only top results of a search are browsed by a user. If interesting information is not found there, a new query may be submitted or the task may be abandoned. A second related problem is that users typically issue poorly defined queries of very few terms. For example, for the query "`java programming`", a user may be interested in tutorials, while another may be interested in source code. This ambiguity in the query is further amplified by the existence of synonyms and homonyms. Synonyms are two words that are spelt differently but have the same meaning. Homonyms are words that are spelt the same but have different meanings. For example, for the query "`apple`", some users may be interested in documents dealing with "`apple`" as "`fruit`", while other users may want documents related to Apple computers. Consequently, without prior knowledge, there is no way for the search engine to predict user interest from simple text based queries.

The above situation gave rise to the idea of personalized search. In particular, storing user preferences in user profiles gives a system the opportunity to return more focused personalized (and hopefully smaller) answers. The general architecture of a personalized search system is depicted in Fig. 1.

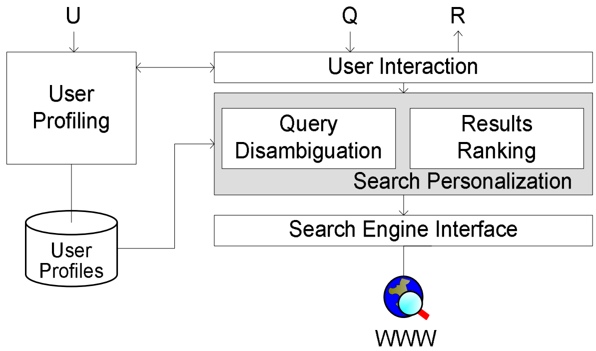

Fig. 1. Architecture of a Personalized Search System

The system keeps a repository of user information (*User Profiles*) that is either inserted explicitly by the user or collected implicitly by monitoring user interaction with the system (*User Profiling*). The user interacts with the digital library through a *User Interaction* component, issuing keyword queries (*Q*) and then browsing the content retrieved (*R*). A query may not represent a unique information need, resulting in generation of many irrelevant answers. For example, a user searching for information on the Java programming language may submit the query "`Java`". The search personalization module may be on top of a traditional search engine or may be integrated into it. The primary ways to personalize a search for an active searcher are *query disambiguation* and *results ranking*. Query disambiguation is typically performed by adding more terms in a query (query augmentation). For example, information that one usually asks about "`programming`" may be recorded. As a result, the query "`Java programming`", which is closer to the actual user information need, is produced. Results ranking comprises re-ordering results returned by the underlying search engine based on user preferences. For example, instead of modifying the initial user query, the information about "`programming`" may be used to place results regarding "`Java programming`" on top of the results returned.

Contributions. Our work concerns keyword searches over unstructured data. We provide a term-based user profile that treats personalization and query disambiguation as a unified term rewriting process (Section 3). Its unique feature is the representation of connections between terms expressing possible rewritings between them. Based on this kind of user profile, we describe a query rewriting algorithm for query disambiguation and personalization (Section 4). Furthermore, results may be ranked based on the proposed user profile. Our framework is independent of the underlying search engine and of the profiling method employed (of course, user profiling effectiveness is an important ingredient for the success of a personalized system). Section 5 provides an overview of a prototype system implementing the proposed framework and discusses user profiling and implementation of searching using Google as the underlying search engine. Experimental results show the potential of the proposed framework.

2 Related Work

Relationships of our work with previous research efforts are sketched below:

Information Retrieval and Filtering. Traditional Information Retrieval systems return the same results to all users issuing the same query [12]. Query disambiguation techniques are used in many of these systems. However, most of them aim to discover corpus-wide word relationships based on co-occurrence analysis of a whole collection (e.g., term clustering [15], similarity thesauri [11]) and they do not take into account how a person perceives word relationships. Information filtering systems employ content-based and/or collaborative filtering methods to return items interesting to a specific user according to a profile capturing long-term user interests [1, 2, 4, 7, 17].

Personalized Searches. Recently Personalized Search systems have emerged. Recall that the primary ways to personalize a search are query augmentation and results ranking. Most current approaches deal with results ranking. For example, Casper [14] ranks jobs returned to a user based on a user profile that specifies job cases previously ranked by the user. Inquirus [6] uses profiles that contain preferences about source selection and results ranking as well as terms from a predefined set that may be inserted to a query. METIORE [3] sorts an answer in the order of user preferences, giving the most interesting solution at the beginning. Their approach of personalization is based on the concept of objective. The user specifies a search objective for every new session. The result of the ranking algorithm is the degree of relevance of an object to the present objective of a user. Persona [16] re-ranks results returned by the underlying search engine based on adapted gradient ascent HITS. [9] presents results for each query under appropriate categories deduced from profiles stored. We differ from these approaches in that we employ user profiles to personalize a user search either at the level of results ranking or at the level of query augmentation. Outride [10] also performs query augmentation. Personalization based on query modification has been also proposed for structured queries over database systems [8].

User Modeling. User modeling refers to representation of user characteristics. In information filtering systems, common user profile representations are borrowed from Information Retrieval and include Boolean, vector-space, and inference models [4, 17]. Outride [10] employs user profiles based upon the ontology of the Open Directory Project (ODP), where each user has his own weighting across the top 1000 categories. In [9], a user profile consists of a set of ODP categories and for each category, a set of terms (keywords) with weights. The weight of a term in a category reflects the significance of the term in representing the user's interest in that category. In [3], for each objective all the documents evaluated are kept along with their evaluation. Each document has some representative features (keywords, author, year, etc…). These inherit the evaluation of the document that contains them. In Persona, the user profile is essentially a mapping of contexts to sets of ODP nodes [16]. A context is defined as a user query. We differ from the aforementioned approaches in that we provide a finer-grained user model which contains weighted terms and connections between them that represent term rewriting operations rather than semantic relations. Different users may have different profiles that do not adhere to some generally accepted concept hierarchy. We believe that the main advantage of our approach is that we can model a user more precisely using terms and term associations that are useful to him. Moreover, this model permits the implementation of both query disambiguation and personalization as a unified process, based on a user profile that records what kind of operations should be performed in order to disambiguate and personalize a user query.

User Profiling. An important building block of a personalization approach is the collection of information about the user to generate a profile (based on the user model). User profiles for information filtering or personalized search are built either manually [6] or with the help of learning techniques [3, 9]. However, the latter typically build profiles consisting of one or more flat term vectors, while the user profile described in this paper is more complex. We describe a simple incremental algorithm for constructing such structured profiles, which we have used in order to evaluate the potential of the approach proposed.

3 User Profiles

We consider queries that are formulated as combinations of terms with the use of logical operators. A term may be a word, e.g., "`Java`", or a phrase, e.g., "`Artificial Intelligence`". We use t_1, t_2, … to denote terms. Logical operators include `AND`, `OR`, `NOT`[1] with their typical semantics. Whenever the operator among keywords is omitted, most search engines consider the logical-`AND` as the default operator. Examples of queries considered are the following:

"`Java AND programming`"

"`geological AND phenomenon`"

Parentheses are used to allow nesting of operators and formulation of complex queries. For example:

"`geological AND (phenomenon OR formation)` "

In practice, it has been observed that queries contain very few terms (one or two). That is why searches are often very ineffective. Given a user query, such as "`Java`", query terms may be combined with other terms through logical operators to produce a query, such as "`Java AND programming AND ZPress editions`", which is less ambiguous and represents user preferences. In this case, disambiguation and personalization of a query may be both viewed as two sides of the same coin, i.e. as a unified term-rewriting process. Hence, we propose a term-based user profile that represents connections between terms based on possible rewritings between them. Modification of a user query is dictated by the user profile. In particular, we model the profile of each user as a directed graph **G(V, E)** (**V** is the set of nodes and **E** is the set of edges) with the following characteristics:

- Nodes in **V** represent terms. The set of terms may be formed based on users' past interaction histories, ontologies, etc. A weight may be assigned to a term, indicating a user's interest in it. Weights are real numbers in the range [`0`, `1`], where a value of `1` indicates extreme interest, a value of `0` indicates lack of interest, and any inter-

[1] Most search engines interpret this binary operator as equivalent to AND NOT.

mediate value indicates an intermediate level of interest for the term on the part of the user. Node weights may be used for ranking of results.

- Edges in **E** represent connections between terms. An edge from term t_i to term t_j may be associated with a specific logical operator and expresses a possible rewriting of t_i using t_j. More specifically, $\mathbf{E}=\mathbf{C} \cup \mathbf{D} \cup \mathbf{N} \cup \mathbf{S}$, where:
  - **C** is a set of *conjunction* edges. A conjunction edge $t_i \rightarrow t_j$ indicates that t_i is rewritten as t_i AND t_j.
  - **D** is a set of *disjunction* edges. A disjunction edge $t_i \rightarrow t_j$ indicates that t_i is rewritten as t_i OR t_j.
  - **N** is a set of *negation* edges. A negation edge $t_i \rightarrow t_j$ indicates that t_i is rewritten as t_i NOT t_j.
  - **S** is a set of *substitution* edges. A substitution edge $t_i \rightarrow t_j$ indicates that t_i is replaced by t_j. This operation is useful for dealing with cases of term misuse.

Table 1 summarizes the edge types, their semantics, and graphical representation.

Table 1. Different edge types and their semantics

Edge Type	Semantics	Notation
conjunction	Given t_i, consider t_i AND t_j	
disjunction	Given t_i, consider t_i OR t_j	
negation	Given t_i, consider t_i NOT t_j	
substitution	Given t_i, consider t_i	

A weight may be assigned to each edge expressing the significance of the specific rewriting of the term for disambiguation/personalization of a query containing it. As with nodes, weights are real numbers in the range [0, 1] with the corresponding interpretation. Fig. 2 provides an example for each edge type. Edge weights are tagged to the edges. Nodes and edges with a weight equal to zero are not stored in a profile.

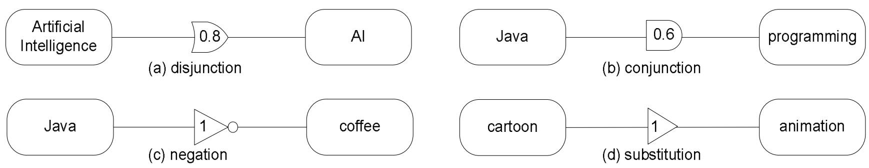

Fig. 2. Examples of edge types between terms

An example user profile is depicted in the graph of Fig. 3. Note that this may be a disconnected graph with components corresponding to unrelated user information needs. For example, in Fig. 3, user interests include background images and Java programming which are unrelated to each other. On the other hand, a connected graph component may capture more than one user need. Based on the profile of Fig. 3, user needs include Java programming as well as database systems, which are connected.

In general, for a pair of nodes t_i and t_j, there may be *at most* one edge $t_i \rightarrow t_j$ whose type would capture the rewriting that best reflects the typical conception of t_i with respect to t_j by the user. On the other hand, both $t_i \rightarrow t_j$ and $t_j \rightarrow t_i$ may exist.

For example, Fig. 3 depicts that "Java" is connected to "programming" and "programming" is connected to "Java". The weights of these rewritings, however, are not the same, since "programming" is also connected to "C".

In addition to immediate term rewriting expressed by an explicit directed edge $t_i \rightarrow t_j$, term rewriting may also be defined transitively by a set of adjacent edges connecting t_i to t_j through intermediary nodes in the profile graph. In that case, t_i may be rewritten using t_j by successively applying the rewritings expressed in the edges. The weight of a transitive rewriting is calculated as a function of the weights of the edges on the corresponding path. Specifically, if D_N is the set of weights along the path, then the transitive weight is expressed as a function $f_T(D_N)$. In principle, one may conceive of different functions that may play this role. Based on human intuition and cognitive evidence [13], these should satisfy the following basic condition:

$$f_T(D_N) \leq \min(D_N) \quad \textbf{(1)}$$

In our prototype, we used multiplication of weights for f_T. For example, for the profile shown in Fig. 3, the weight of the transitive rewriting of "Java" using "database systems" is 0.9*0.5=0.45. Another possible function is the minimum of weights. For the same example, the corresponding transitive weight would be 0.5.

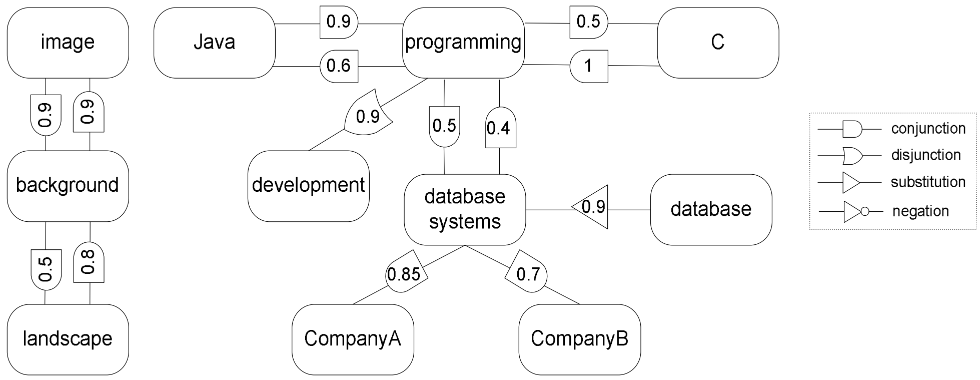

Fig. 3. Example user profile

It should be pointed out that not all graphs with the above characteristics map to valid profiles. For example, between two terms there cannot be substitution edges in both directions. In addition, when constructing a profile, it must be determined which terms are connected with edges and which terms are connected through other terms.

A separate profile as described above must be kept for each individual user of a system. Moreover, its overall design is quite general, so it can capture in exactly the same way profiles of groups of users as well but this is out of the scope of this paper.

4 Query Disambiguation

Based on the above, disambiguation and personalization of a user search are viewed as a unified query modification process, consisting of term-rewriting operations dictated by the user profile. An algorithm for this purpose is presented in Fig. 4, and is called *QDP* (*Query Disambiguation and Personalization*). The algorithm takes as inputs a user query Q, a user profile U and a criterion CXT and produces a single modified query Q′. The criterion CXT establishes the *query context*, i.e. it specifies which terms of U, connected (directly or transitively) to terms included in Q, influence the original user query and, should therefore be considered in query modification. Examples of possible criteria are the following:

– The transitive weight of any path connecting a new term t_j to the query Q should be greater than a threshold T.
– The number of edges of any path connecting a new term t_j to the query Q should be less than a threshold T.

For example, for the query "database systems" and the profile of Fig. 3, consider the first criterion with a threshold T=0.6. Then, the terms "CompanyA" and "CompanyB" are the only ones considered within the query context. Criteria may be manually specified by an expert or automatically configured by the system.

The algorithm is based on a best-first traversal of the graph representing the user profile. The basic idea is to gradually modify the query by considering, in each round, terms from the profile that are connected to those already in the query. The algorithm stops when there are no terms in the context of the original query to be used for further query refinement.

```
QDP algorithm
Input:   Query Q, User Profile U, Query-Context Criterion CXT
         /*CXT(tj,Q)=TRUE then tj is in the context of Q */
Output:  Modified query Q'

Begin
Q'=Q
Pick (ti→tj)∈U s.t.:
  ti∈Q', tj∉Q', ti→tj has max. weight among rewritings of Q'
While CXT(tj,Q)=TRUE
     Replace ti∈Q' with the result of rewriting (ti→tj)
     Pick (ti→tj)∈U s.t.:
     ti∈Q' and ti→tj has max. weight among rewritings of Q'
End
```

Fig. 4. Outline of *QDP* algorithm for personalization

More specifically, the original query is mapped to the user graph. Any query term not mapped to a node in the profile is not affected by the algorithm. In each round, a term t_i from the current version of Q′ is selected provided that there is an edge $t_i \rightarrow t_j$ stored in the profile with the greatest weight among all possible rewritings of any term in Q′ such that t_j is within the context of the original query according to criterion CXT and is not already included in Q′. Query modification stops when there are no terms in the context of the original query that can be integrated into it. Fig. 5 illustrates successive transformations of a query for the profile illustrated in Fig. 3 using this algorithm with a threshold T=0.8. A slightly different version of *QDP* is employed if there are more than one possible rewritings of a specific term, such as "Java" connecting to "programming" and "island". In this case, the algorithm may return a set of queries. Depending on the personalization philosophy adopted by the system, the user may be provided with this set in order to choose which one better represents their current need, or, with the top-ranking results returned by each query. Details are omitted due to space considerations.

Java → Java AND programming → Java AND (programming OR development)

Fig. 5. Execution of `QDP` using a query "`Java`", the example profile and a threshold `T=0.8`.

5 Implementation

In this section, we provide an overview of the prototype system that we have built following the architecture of Fig. 1 and the proposed framework. We have built this system on top of an existing search engine (Google).

Search Interface. We have used the Google Web API service, a beta web program that enables developers to find and manipulate information on the web [18]. A personalized query, output by *QDP*, is translated into the appropriate syntax format expected by Google based on certain rules. For example, search for complete phrases is performed by enclosing them in quotation marks. A word is excluded from the search (due to a `NOT` operator dictated by the profile) by placing a minus sign immediately in front of it. Results by Google are displayed to the user. We are implementing a module that re-ranks results based on term weights stored in the user profile.

User Profiling. We implemented a simple incremental algorithm for construction of profiles based on the user model presented with the purpose of evaluating the potential of the user model and of the query modification process. We only sketch it here. It builds profiles with no negations based on user feedback and one-keyword queries.

When a user is presented with the results of a search, he may mark relevant documents. These are used as input to the profile construction algorithm which builds a sub-graph whose structure depends on the documents input as well as the number of them. The algorithm proceeds in three steps. First, document analysis is applied. Each document is mapped to a list of {`word, number of word occurrences`} pairs. Then, these lists and the initial query are used to construct a sub-graph. Initially, this sub-graph consists of one node corresponding to the query term. The algorithm gradually builds the sub-graph by adding nodes mapping (groups of) words encountered in the input document lists. Two or more words are treated as one group if the algorithm cannot decide on the number and type of edges to be used to connect them if these words are mapped to separate nodes. Rules for creation of edges include:

```
The query term and a group of words co-exist in a given list ⇒
A conjunction edge is created between the respective nodes
The query term does not exist in a given list ⇒
A disjunction edge is created between the query node and the node(s) map-
ping the words of this list.
The query term does not exist in any given list ⇒
A substitution edge is created between its respective node and the node(s)
```

mapping the words of the lists.

At the end of this step, nodes in the sub-graph represent terms and edges represent term rewritings. Finally, the algorithm merges the sub-graph produced in the previous step into the user's profile and assigns node and edge weights. If the user profile is empty, or has no similar nodes with the sub-graph, then the latter is simply added as a disconnected graph into the former. Similar nodes are those sharing common words, such as nodes $\{t_1\}$ and $\{t_1, t_2\}$. If the profile and the sub-graph have similar nodes, then they are merged. Merging preserves finer-grained structure which may already exist in the profile or dictated by the connections in the sub-graph. Each node of the profile that is new or has been affected by the merging is assigned the minimum of weights of the words mapped to it. The weight w_t of a word is given by the formula:

$$w_t = f_t * (N_t/N) \quad \textbf{(2)}$$

N_t: the number of past and current documents containing the word; N: the total number of documents of sets containing at least one document with this word; f_t: the number of the word's past and current occurrences to the total number of words of sets where the word exists. An edge is assigned a weight w_e with a value equal to:

$$w_e = (N_e/N') \quad \textbf{(3)}$$

N_e: the number of past and current documents containing associated words; N': the total number of documents of sets with at least one document including these words.

Experimental Results. We have conducted experiments with ten users. Since the underlying engine is Google, our dataset is the Web. Each user was assigned three search tasks, such as finding an appropriate car rental, identifying the latest in fashion, and so forth, as well as allowed to consider two tasks of their own. The goal of a search task was to find at least two interesting and relevant resources. Each user had two sessions with the system. During the first one, no modification of user queries took place. The user could resubmit a query until the goal of a task was satisfied. In addition, users submitted feedback by marking relevant documents. This information was used by the profiling method. During the second phase, which took place three weeks after the first one, personalization was activated using the profiles built during the first session. As the main measure of effectiveness of our approach we have used the task completion time. On the average, participants satisfied their needs significantly faster when personalization was in effect rather than when queries were executed in their original form. Specifically, the average gain in time was equal to 29%. In addition, there was an increase in the number of relevant documents found among the top 20 results returned by the search engine when personalization was applied. In some cases, when the initial user query had been extremely ambiguous, improvement was over 50% (i.e. over half of the top 20 results have been replaced by more relevant matches).

6 Conclusions and Future Work

In this paper, we have described a high-level architecture of a system for personalized searches. We have described a term-based graph-form user profile that allows thinking of query disambiguation and personalization as a unified term rewriting process. It interprets connections between terms as possible rewritings between them. Based on this kind of profile, we have provided a query-rewriting algorithm for query disambiguation and personalization. We have described a prototype system and presented promising results regarding the potential of the proposed framework.

We are currently elaborating the prototype system. We are working on the ranking module, and on a more elaborate user profiling algorithm capable of handling negations and initial queries with more than one term. We are also interested in developing a mechanism for user profile validation and in more extensive experiments.

References

1 Belkin, N., Croft, W.B. (1992). Information Filtering and Information Retrieval: Two sides of the same coin?. *Communications of the ACM*, Vol.35, No.12. December 1992.

2 Bollacker, K., Lawrence, S., Giles, C. (1999). A System for Automatic Personalized Tracking of Scientific Literature on the Web. Int'l. Conf. on Digital Libraries, 105-113, 1999.

3 Bueno, D., David, A. (2001). METIORE: A Personalized Information Retrieval System. M. Bauer, P.J. Gmytrasiewicz, and J. Vassileva (Eds.): UM 2001, LNAI 2109, 168–177, 2001.

4 Callan, J. (1996). Document Filtering with Inference Networks. ACM SIGIR Conf., 262-269, 1996.

5 Cetintemel, U., Franklin, M., Giles, C. (2000). Self-Adaptive User Profiles for Large-Scale Data Delivery. Intl. Conf. on Data Engineering, 2000.

6 Glover, E. J., Lawrence, S., Birmingham, W. P., Giles, C. (1999). Architecture of a Metasearch Engine that Supports User Information Needs. Int'l. Conf. on Information and Knowledge Management, 1999.

7 Herlocker, J., Konstan, J., Borchers, A., Riedl, J. (1999). An Algorithmic Framework for Performing Collaborative Filtering. Intl. Conf. on Research and Development in IR, 230 - 237, 1999.

8 Koutrika, G., Ioannidis, Y. (2004). Personalization of Queries in Database systems. Intl. Conf. on Data Engineering, 597-608, 2004.

9 Liu, F., Yu, C., Meng, W. (2002). Personalized Web Search by Mapping User Queries to Categories. Intl. Conf. on Information and Knowledge Management, 2002.

10 Pitkow, J., Schutze, H., Cass, T., Cooley, R., TurnBull, D., Edmonds, A., Adar, E., Breuel, T. (2002). Personalized Search. *Communications of the ACM*, Vol. 45. No.9, Sep. 2002.

11 Qiu, Y., Frei, H.P. (1993). Concept-based Query Expansion. Intl. Conf. on Research and Development in IR, 160-169, 1993.

12 Salton, G., McGill, M.J. (1983). Introduction to Modern IR. McGraw-Hill, NY, 1983.

13 Collins, A., Quillian, M. (1969). Retrieval Time from Semantic Memory. *J. of Verbal Learning and Verbal Behaviour*, Vol 8, 240-247, 1969.

14 Smyth, B., Bradley, K., Rafter, R. (2002). Personalization Techniques for Online Recruitment Services. *Communications of the ACM*, Vol. 45, No. 5, 39-40, May 2002.

15 Sparck Jones, K., Jackson, D. M. (1976). The use of automatically-obtained keyword classifications for Information Retrieval. Information Processing and Management, 1976.

16 Tanudjaja, F., Mui, L. (2002). Persona: A Contextualized and Personalized Web Search. Proc. of the 35th Hawaii Int'l Conf. on System Sciences, 2002.

17 Yan, T., Garcia-Molina, H. (1995). SIFT: A Tool for Wide Area Information Dissemination. USENIX Technical Conference, 177-186, 1995.

18 Web API for Google. http://www.google.com/apis/api_faq.html

Towards Mobile Tour Guides Supporting Collaborative Learning In Small Groups

Michael Kruppa[123], Andrew Lum[3], William Niu[3], and Miriam Weinel[3]

[1] International Post-Graduate College Language Technologies and Cognitive Systems, Saarland University, Saarbrücken, Germany
[2] DFKI GmbH, Saarbrücken, Germany
[3] University of Sydney, Australia
{michaelk, alum, niu}@it.usyd.edu.au
m.weinel@edfac.usyd.edu.au

Abstract. Within this paper we address the problem of supporting small groups of museum visitors by means of museum tour guides based on mobile devices. Instead of forcing members of such groups into isolation, as today's personal museum guides do, we aim at supporting the potential of collaborative learning within such groups. In order to evaluate our approach, we propose a large scale user study evaluating the effectiveness of the group collaboration encouraged by the system.

1 Introduction

Starting with simple, audio-cassette based museum tour guides, the development of such guides over the past years has led to mobile museum tour guides capable of automatically reacting to the user context (i.e. user position and orientation) and automatic adaptation to users specific preferences and needs (e.g. [1]). Some of the systems are even capable of dynamic content adaptation, based on the user's interaction with the system (e.g. [2]). Furthermore, modern mobile museum tour guides are no longer limited to audio output. They feature mobile devices with high resolution colour screens, allowing for images and video-clips to be watched on the device. Even virtual characters, playing the role of a museum tour guide (see [3]), have been realised.

However, all of these different mobile tour guides share a common concept. They are designed to support an individual, single visitor in a museum setting. The original idea of the audio-cassette museum tour guides was to allow individual visitors of a museum to have a guided tour, since human tour guides are usually only available for groups. However, most of the visitors in museums nowadays do neither come as a large group (allowing them to have a human guide) nor as an individual. They visit the museum in small groups, like a family or a couple of friends (as discussed in [4]). Using any of the currently available personal, mobile tour guide systems will force the members of these small groups to separate and take a tour designed for individuals. Instead of sharing a common experience and benefiting from each others knowledge and understanding of the material presented in the museum, the individual group members become isolated.

The work described in this paper will aim at the support of these small groups. We envision a situation where the following scenario becomes possible:

Charles, Julia and Ben are visiting a technically oriented museum. While Charles is interested in electronics, Julia and Ben are more into mechanics. Each of them is renting a mobile museum tour guide. The system, being aware of the individual user interests, automatically determines an optimal tour through the museum which will be of high interest to each of them. While visiting the individual exhibits suggested by the system, different content, focusing on the particular interests of each group member, is delivered to each of them. However, these presentations are not completely individual, since they share some common parts (i.e. general information) and even more, since the group members visit the same exhibits together. For example, when looking at a plane in the museum, Charles will receive detailed information on the navigation systems while Julia and Ben are informed about the mechanics of the under carriage. However, all of them will first get some introductory information, for example on the manufacturer of the plane, what it was used for and how it came to end up in the museum. After a presentation regarding a specific exhibit is over, the system calculates the potential knowledge gain in each group member and encourages them to share what they have learned with their friends. At the end of the visit, Charles, Julia and Ben have had a shared experience in the museum while their individual interests have been satisfied.

The basic idea behind our concept is to support the collaborative learning potential within small groups when visiting a museum. We plan to integrate virtual characters in order to facilitate the communication between group members, since literature suggests that these characters have a positive influence on the effectiveness in computer based learning scenarios [5]. We believe, that our approach based on mobile devices yields a high potential to support group collaboration [6].

We plan to conduct a large scale user study ($\sim$ 100 participating individuals) with children in the age group of 10-12 years. The goal of the user study is to evaluate the proposed mobile museum guide with respect to the task difficulty perception and knowledge gain of the participants as well as the effectiveness in supporting group collaboration within small groups.

In the following section, we will discuss the theoretical background of our approach, which is followed by a detailed overview on the set up and proposed evaluation method of the user study.

2 Theory and Background

2.1 Informal Mobile Learning in Museums

Mobile technologies have become very popular over the last 10 years. They emerged in a variety of different aspects of human life, e.g. mobile phones changed the way we communicate. Since the late 1990s increasing interest has emerged in mobile technology usage within educational settings [7].

During the last five years mobile learning has been applied to a variety of educational settings, e.g. outdoor learning, learning in museums, classroom settings. Mobile learning incorporates two major advantages over other forms of technology-supported learning. First of all, mobile technology is not limited to one specific place, thus providing a greater integration into learning activities and the surrounding environment [8]. Learning used to be restrained to the place where the computer station was set up, now the learner is independent of location. Moreover increased mobility supports face-to-face collaboration among children [9]. Now children not only have to interact virtually with each other at another computer station but also can interact with each other directly, yet having digital tools at their disposal. Secondly, mobile technologies are context- as well as location-sensitive [10]. They are aware of their own location as well as that of other devices. Context sensitiveness is of interest for educational settings where learning itself is spread throughout space and specific collaboration is thought to take place among learners. This has proven especially valuable for places where learning takes place outdoors [11] like national parks. But it has also shown to have a great impact on learning that occurs indoors: in classroom settings, where it solved problems in settings without technological support [6], as well as in informal learning settings, like museums for instance.

Usually studies on mobile learning in museums involve a mobile device displaying a guided tour of the exhibitions in that particular museum. The guided tour in general is multimedia based, incorporating pictures, text, audio and/or video.

Still, there are some limiting aspects as well. The majority of studies are realised with PDAs and they do not allow for much content to be shown at once [8]. While evaluating learning scenarios, one should consider that tablet Pcs might be more difficult to handle but they also provide the required functionalities to engage in collaborative knowledge building like concept mapping. Moreover there are new emerging demands that ask for further technological and software developments like lower energy consumption for hardware or improved user interfaces [12]. Even though there still seems to be a need for further improvements, technological as well as pedagogical, quite a number of researchers are convinced of the future importance and impact of mobile technologies in education [13]. We share this belief, but would argue that in particular with catering to the needs of younger learners, more will need to be done with mobile devices than just making instructional content available to them. Virtual characters, we argue, are particularly well suited to support the segment of young and very young learners in informal settings.

2.2 Informal Mobile Learning with Virtual Characters

Research on virtual characters has been conducted primarily from a technological perspective [14]. A few studies started to consider the usage of virtual characters in educational settings [15].

One advantage of virtual characters is their social nature. Even though studies address the beneficial impact of virtual characters on learning [15], they have not been tied up with mobile learning environments on a larger scale. There is some evidence that the mere presence of a virtual character results in perceptions of reduced task difficulty [16]. This has been coined the persona effect. Since mobile learning fosters social interaction as well as mobility, it seems to be especially important to provide a guide or model that leads the learner through the learning process. Virtual characters can provide the support needed by learners during learning activities [17]. Some research has been done on the social effects of virtual characters in interactions with humans [18]. Relating those effects to human learning in informal mobile learning environments is yet to be done.

Providing the learner with a variety of perspectives on the learning content promotes knowledge construction [19]. The provision of learners with multiple perspectives is one of several principles stated in constructivist learning theories and is said to support and enhance the learning process. Some evidence can be found that the usage of virtual characters in constructivist learning environments results in greater knowledge acquisition [20].

In the current study we would like to address the above stated research lack through introducing virtual characters to an informal mobile learning environment that is based on constructivist principles. Thus combining advantages of mobile learning encouraging collaborative learning in museums with the implementation of social supportive aspects of virtual characters.

2.3 Group Modelling

As museums are more in tune with their social values and contemporary issues, they are increasingly recognised as social places. This is reflected in the work of Petrelli et al. [4], who observed that very few visitors went to the museum alone. It is conceivable that a group of people may have conflicting preferences and needs, and the generation of a recommendation which addresses the requirements of all the people in the group is much more complex than that of an individual. Moreover, they also observed that people act differently when visiting museums with companions, as opposed to visiting alone. People's behaviour typically depended on the group of people they were with (e.g. a person may be more motivated to learn about how a motor operates when he is with a knowledgable friend).

Much work has been done in addressing groups in various domains. Masthoff [21] conducted two experiments on how a group of people select a series of TV programs for a group of viewers. She explored how people choose a sequence of programs to watch based on individuals' preferences from a third-person poitn of view. She also examined the satisfaction level people indicated with sequences of programs applied with different strategies. The results show that people (1) accounted for fairness and tried to avoid individual misery, (2) used normalisation (i.e. their satisfaction is based on both selected and non-selected items), and (3) used ratings in a non-linear way.

Another example of a group modelling system is MusicFX [22] used in a fitness centre to adjust the selection of background music to best suit the preferences of the people working out at any given time. One interesting facet of the system is that a group in the context is made up of people who happen to work out at the same time, unlike the sense of a group addressed in most other projects (e.g. [21], [23], [24]), who are non-strangers, such as friends or family members. Another distinctive aspect of the system is that it uses explicit preferences of all inhabitants to make a selection that will directly affect everyone who is present. The evaluation demonstrates that a good majority of users (71%) felt the music being played improved and that the people's musical preferences were better matched.

Group modelling is also attended in city tours. In the INTRIGUE project, the ranking of attractions are separately ranked by first considering the preferences of each *homogeneous subgroup* before combining them to obtain the overall ranking for the whole group [25]. In addition, some subgroups could be particularly influential either because it contains a majority of the members of the group or because it represents a relevant tourist class (e.g. children and disabled people).

Although a fair amount of literature has contributed to modelling heterogeneous groups in various domains, yet there is little work in the museum settings. Sotto Voce [24] is designed to accommodate a group of users, but does not include a personalisation component. The PEACH project also raised some interesting issues in how to personalise the information delivered, for example, following Kruppa [23] in providing some common information to the group on a large display and some personalised information on a hand-held display. Kay et al [26] also presented a scenario of how an adaptive museum guide may deliver personalised information to each individual in a group, which in turn stimulates after-visit group discussions.

3 Proposed Experiment

In order to evaluate the proposed system, we are designing a user experiment, which will be carried out in cooperation with the Nicholson Museum[4], the first Australian archeological museum, located at the University of Sydney campus. The goal of the user study is to evaluate the effect of the proposed mobile museum guide in supporting group collaboration within small groups. By running this comparative experiment, we expect that the groups accompanied by a virtual character will prevail. In other words, participants with a virtual character are expected to have lower perception of task difficulty and higher knowledge gain than those without a virtual character.

[4] The Nicholson Museum: http://www.usyd.edu.au/nicholson/ (accessed 28 February 2005).

3.1 Settings and Methods

This study will take place during regular school-class visits in the Nicholson Museum. The museum provides a couple of programs[5] designed to complement the Primary and Preliminary Syllabus[6] in the New South Wale, Australia. We would like to target our subjects to primary children aged 10–12. We expect to have around 100 participants.

We envision the experiment to be of 2x2 research design, as shown in table 3.1. Each participant of the visit groups will belong to one of the four divisions, which will be around 25 participants in total for each of the conditions. To begin with, half of the participants in each visit group will be presented with a mobile tour guide system on a tablet PC (described in Section 4) with a virtual character to browse the museum, and the other half will have one without a virtual character. After the exploration phase, the system will automatically analyse the potential knowledge gain and implicitly partition each group into experts and novices. Before asking the participants to fulfill some collaborative tasks, the system will further organise the participants in groups of three or four with different expertise in certain areas. After completing the tasks, each participant is presented with a questionnaire inquiring aspects on learning effectiveness.

	With Virtual Character	Without Virtual Character
Experts	25 subjects	25 subjects
Novices	25 subjects	25 subjects

Table 1. Research design

4 System Architecture

The system we plan to build consists of four main components - the user modelling server, the ontology, the virtual character interface, and the adaptive hypertext interface. The latter will effectively render the same museum content with and without the virtual character respectively. Fig. 1. shows a diagram of the system and how they interact with each other. Further descriptions of each component are described below.

4.1 Ontology

We propose to model the museum context with a light-weight ontology. An ontology describes not only the concepts but also the relationships between concepts

[5] The program pamphlet can be downloaded at: http://www.usyd.edu.au/nicholson/nichol_primary.pdf (accessed 05 March 2005).

[6] The syllabi and relevant resources can be found at the Web site of the Board of Studies, New South Wales, Australia: http://www.bosnsw-k6.nsw.edu.au/index.html (accessed 05 March 2005).

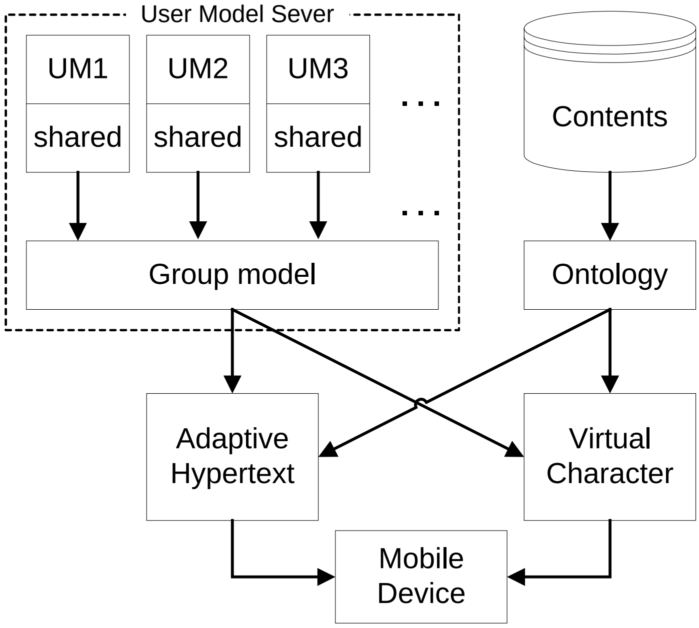

Fig. 1. Proposed system architecture showing the main components.

in a domain, and thus forms an ideal schema for semi-structured data. Museums commonly have small descriptions to accompany each exhibit. If all these exhibit descriptions were collated together, we would have, in effect, a specialised glossary of the museum contents. It would therefore be very useful to have an automated means to generate a light-weight ontology from these descriptions.

MECUREO [27] is one such tool to fit this task, designed to automatically construct ontologies from glossaries and dictionaries. MECUREO can not only create an ontology from these descriptions, but also one that is scrutable and understandable as every concept leads back to a description.

The vocabulary of the ontology forms the basis for the domain concepts in the user model, and the relationships can be exploited by the resolvers in user modelling systems to perform intelligent customisations. Through examining these ontological relationships, the resolvers can infer related concepts and provide recommendations to the users based on these inferences.

4.2 User Model Server

The personalised information delivered to museum visitors is powered through a user model. Actions by the user are stored in the user model as evidence to allow the system to create adaptations to enrich their experience of the visit. The evidence would range from the duration they have spent viewing or interacting with a particular exhibit to the history of exhibits they have visited.

In order to adapt information for a heterogeneous user group, a number of approaches is available. One is to build a group model from the individual models; an alternative approach is to interpret the individual models to take account of the interactions between the people who constitute the group [28]; yet another approach is to model each *homogeneous subgroup* extracted from the group, as proposed in [25]. The relative merits of either approach need to be explored.

We plan to use the user modelling server Personis [29], which allows adaptive systems to easily manage evidence for user models. Personis provides a resolution system to perform customisations based on this evidence. The resolver API allows system designers to easily code in adaptation rules based on accreted evidence and can be accessed by different devices and applications and the results can be exported to various file formats such as XML.

4.3 Virtual Character

The virtual characters we will use throughout the experiment will be adopted from previous project work (described in [3]). These characters were developed in Macromedia Flash MX[7] and are hence web based. This will allow for an easy integration of the html based content representation and the virtual character. The characters feature two different layouts (a minimal one for small screen devices like a PDA and a full sized one for large screen devices). The characters are capable of performing utterances by means of synchronised mp3 playback. The character engine is script driven and remotely controllable by a server over the network.

4.4 Adaptive Hypertext

We have been developing a version of the TUTOR system [30] that integrates the Personis user modelling server. The web-based interface renders content that is personalised for each individual user. Support for group modelling exists through the aforementioned Personis backend. The system has been developed in the domain of undergraduate computing courses, but recent work has seen it adapted to a museum context that we can build on for this user study.

5 Summary

In this paper we presented the concept and theoretical background of a proposed mobile museum tour guide which will support small groups of visitors in a museum. Even though small groups of visitors are very common in museums nowadays, they are seldom explicitly supported by mobile museum tour guides. In doing so, we hope to unleash the potential of collaborative learning within the group. The proposed initial experiment will evaluate the technological support of this collaborative learning potential by means of virtual characters on mobile devices. Based on the experiences gained during our experiment we hope to build a museum tour guide system which will improve the learning experience and enjoinment of each group member during their museum visit. Instead of breaking

[7] http://www.macromedia.com

these groups apart, our system will encourage communication and collaboration among the single group members, resulting in a beneficial, shared experience for the whole group.

Acknowledgements

This project is partially funded by Smart Internet Technology CRC and Hewlett Packard.

References

1. Proctor, N., Tellis, C.: The state of the art in museum handhelds in 2003. In: Proceedings of Museums and the Web. (2003)
2. Sparacino, F.: The museum wearable: real-time sensor-driven understanding of visitors interests for personalized visually-augmented museum experiences. In: Proceedings of Museum and the Web 2002, Boston (2002)
3. Kruppa, M., Krüger, A., Rocchi, C., Stock, O., Zancanaro, M.: Seamless personalized TV-like presentations on mobile and stationary devices in a museum. In: Proceedings of the International Conference on Hypermedia and Interactivity in Museums (ICHIM). (2003)
4. Petrelli, D., Angeli, A.D., Convertino, G.: A user centered approach to user modelling. In Kay, J., ed.: Proceedings of the Seventh International Conference on User Modelling (UM'99), Springer Wien New York (1999) 255–264
5. Gulz, A.: Benefits of virtual characters in computer based learning environments: Claims and evidence. International Journal of Artificial Intelligence in Education **14** (2004) 313–334
6. Zurita, G., Nussbaum, M.: Computer supported collaborative learning using wirelessly interconnected handheld computers. Comput. Educ. **42** (2004) 289–314
7. Danielsson, K., Hedestig, U., Juslin, M., Orre, C.: Participatory Design in the Development of Mobile Learning Environments. J. Attewell and C. Savill-Smith (2004)
8. Stanton, D., Neale, H.: Designing mobile technologies to support collaboration. In: Technical Report Equator-02-208. (2002)
9. Danesh, A., Inkpen, K., Lau, F., Shu, K., Booth, K.: Geney: Designing a collaborative activity for the palm handheld computer. In: Proceedings of CHI, Conference on Human Factors in Computing Systems. (2001)
10. Dix, A., Rodden, T., Davies, N., Trevor, J., Friday, A., Palfreyman, K.: Exploiting space and location as a design framework for interactive mobile systems. ACM Transactions on Computer-Human Interaction. (2000) 285–321
11. Chen, Y., Kao, T., Sheu, J.: A mobile learning system for scaffolding bird watching learning. Journal of Computer Assisted Learning (2003) 347–359
12. de Freitas", S., Levene, M.: Evaluating the development of wearable devices, personal data assistants and the use of other mobile devices in further and higher education institutions. In: "JISC Technology Standards Watch Report: Wearable Technology". (2003)
13. Roschelle, J., Pea, R.: A walk on the wild side: How wireless handhelds may change cscl. In: Proceedings of Computer Support for Collaborative Learning. (2003)

14. Clarebout, G., Elen, J.: Animated pedagogical agents: An opportunity to be grasped? Journal of Educational Multimedia and Hypermedia (2002) 267–286
15. Mayer, R., Dow, G., Mayer, S.: Multimedia learning in an interactive self-explaining environment: What works in the design of agent-based microworlds? Journal of Educational Psychology (2003) 806–812
16. Mulken, S.V., Andrè, E., Müller, J.: The persona effect: How substantial is it? In: HCI '98: Proceedings of HCI on People and Computers XIII, Springer-Verlag (1998) 53–66
17. Lester, J., Towns, S., Fitzgerald, P.J.: Achieving affective impact: Visual emotive communication in lifelike pedagogical agents. Computers and Education (2004) 289–314
18. Fong, T., Nourbakhsh, I., Dautenhahn, K.: A survey of socially interactive robots. Robotics and Autonomous Systems (2003) 143–166
19. Oliver, R., Herrington, J.: Exploring technology-mediated learning from a pedagogical perspective. Journal of Interactive Learning Environments (2003) 111–126
20. Moreno, R., Mayer, R.E.: Pedagogical agents in constructivist multimedia environments: The role of image and language in the instructional communication. In: Paper presented at the annual meeting of the American Educational Research Association. (2000)
21. Masthoff, J.: Group modeling: Selecting a sequence of television items to suit a group of viewers. User Modeling and User-Adapted Interaction **14** (2004) 37–85
22. McCarthy, J.F., Anagnost, T.D.: Musicfx: an arbiter of group preferences for computer supported collaborative workouts. In: CSCW '98: Proceedings of the 1998 ACM conference on Computer supported cooperative work, Seattle, Washington, United States, ACM Press (1998) 363–372
23. Kruppa, M.: A user study on parallel presentations for heterogeneous user groups. Technical Report 563, Saarland University (2004)
24. Aoki, P.M., Grinter, R.E., Hurst, A., Szymanski, M.H., Thornton, J.D., Woodruff, A.: Sotto voce: Exploring the interplay of conversation and mobile audio spaces. In: Proceedings of the SIGCHI conference on Human factors in computing systems (CHI '02), Minneapolis, Minnesota, USA, ACM Press (2002) 431–438
25. Ardissono, L., Goy, A., Petrone, G., Segnan, M., Torasso, P.: Intrigue: personalized recommendation of tourist attractions for desktop and hand held devices. Applied Artificial Intelligence **17** (2003) 687–714
26. Kay, J., Lum, A., Niu, W.: A scrutable museum tour guide system. In Butz, A., Kray, C., Kruger, A., eds.: 2nd Workshop on Multi-User and Ubiquitous User Interfaces (MU3I'05). (2005) 19–20
27. Apted, T., Kay, J.: Mecureo ontology and modelling tools. WBES of the International Journal of Continuing Engineering Education and Lifelong Learning **14** (2004) 191–211
28. Stock, O., Zancanaro, C.R.M., Kuflik, T.: Discussing groups in a mobile technology environment. In Buts, A., Kray, C., Kruger, A., Schmidt, A., Prendinger, H., eds.: 2nd Workshop on Multi-User and Ubiquitous User Interfaces 2005 (MU3I 2005) in Conjunction with the IUI'05, San Diego, California, USA, ACM Press (2005)
29. Kay, J., Kummerfeld, B., Lauder, P.: Personis: A server for user models. In: Proceedings of Adaptive Hypermedia and Adaptive Web-Based Systems (AH'02), Springer-Verlag, London, UK (2002) 203–212
30. Czarkowski, M., Kay, J.: A scrutable adaptive hypertext. In: Proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems (AH '02), Springer-Verlag (2002) 384–387

Designing Personalised Information Access to Structured Information Spaces

George D. Magoulas and Dionisios N. Dimakopoulos[1]

London Knowledge Lab and School of Computer Science, Birkbeck College
University of London, WC1E 7HX, UK
{gmagoulas, dionisis}@dcs.bbk.ac.uk

Abstract. The paper presents an approach to personalisation of structured information spaces that builds around a set of services. Structured information representation is increasingly being used to improve the organisation, search, and analysis of information spaces provided on the Web and is very popular in digital libraries, classification-based search engines, information directories, and subject gateways. In a service-oriented approach, the application behaviours contained in the various systems are defined as services which are "open" and can be consumed by other applications. The paper identifies relevant personalisation services, discusses their expected behaviours, and explores the dimensions of individual differences that should be included in a user model specification to meet personalisation services requirements and create personalised information access.

1 Introduction

The concept of information spaces on the Internet spans over various domains, such as hypertext documents, digital libraries, subject gateways, web directories, newsgroups and mailing lists [6]. The diversity, organizational heterogeneity, immense size and dynamic expansion that characterize Web information spaces have made information searching, navigation and browsing quite challenging tasks: (i) some information spaces are not clearly delimited; (ii) users' abilities can vary greatly and their level of domain understanding may grow differently during interaction as it depends on their knowledge background and expertise; (iii) sometimes information services have been developed by content providers without enough thought given to interface design considerations, information presentation and organization; (iv) users may be affected by errors and omissions that were made during construction of the space, as they may experience situations like not being able to locate the information they need or "being misled" when browsing through the search results or the information categories..

This paper focuses on information spaces that adopt a structured information representation approach. In a highly structured virtual information space, all

[1] The authors gratefully acknowledge the financial support of the UK Engineering and Physical Sciences Research Council (EPSRC Grant Reference: GR/R92554/02).

information related to the same topic can be found under the same subcategory. Pieces of information that relate to multiple topics will appear under several distinct subcategories. Most of existing structured information spaces comprise a structure for the organisation of the content, metadata descriptions regarding the semantics of the content and their access properties [28]. In a structured information space personalisation can exploit user metadata and resources metadata. A personalised view of the information space can be created by matching user model attributes with resources attributes. This matching process can be also considered as an inference mechanism, such as those based on rules, to determine whether a service is recommended, identify relevant resources or support user navigation thought the content by generating tailored navigation paths.

Integration of adaptation techniques in existing systems is considered the next challenging step in the evolution of personalised services. To this end, several approaches have been proposed so far, such as those explored in the context of web-based instructional systems [2], e.g. standard-based reusability, resource discovery architectures and semantic web technologies for learning [7]. Towards this direction this paper adopts a service-oriented approach, [32], as a common framework to personalise the access in an integrated information space. This approach facilitates the integration of commercial, in-house and open source components and applications within organisations and regional federations by agreeing upon common service definitions, behaviours, data and user models, and protocols. The rest of the paper discusses a subset of these issues: the next section identifies and defines personalisation services for information spaces. Then aspects of the user are considered and data models for user profiling are discussed to support these personalisation services. The paper ends with discussion and future work.

2 Service-oriented Approach for Personalised Information Spaces

Service-oriented approaches provide several benefits, such as support for planning technical and interoperability specifications and standards development, enable alignment with business processes and support business models, offer flexibility to accommodate evolving organisational requirements, provide a flexible and modular technology base, make information sharing of applications simpler and allow collaborative organisations to deploy applications that meet their common needs.

Service-oriented approaches for personalisation allow the development of modular and flexible personalised systems, [1], where the components can be added, removed or replaced more easily than in traditional models of adaptive hypermedia systems, and where new applications or systems can be composed from collections of available services. They also enable faster deployment of personalisation technologies as long as the needs of new components are compatible with the existing component interfaces. This approach is different from integrating directly at the user interface level (e.g. by using portals) or at the data level (e.g. by creating large datasets or data warehouses). For example, a student record system may provide services for enrolment and registration processes which can also be used by an cross institutional library system to allow registered students to access online course materials and related information resources to collaborating institutions. Another example is a personalised recommendation service that can be utilised by a variety of applications to recommend web content, learning objects to study in a virtual learning environment, or learning opportunities in a professional/personal development system.

A service-based architecture may provide personalisation on the basis of well defined service behaviours and interfaces and allows various open specifications, open source toolkits and standards to be used in implementing the services. From the functional definition and scope of a specific service an abstract model of behaviour and data can be developed, which describe the expected behaviour of a realisation of this service and the data model (e.g. using XML) it deals with or exchanges. A service can be realised in a number of ways, such as a Web service (e.g. using WSDL) and Application Programming Interfaces for particular programming languages. The various services interact to provide the complete functionality [1].

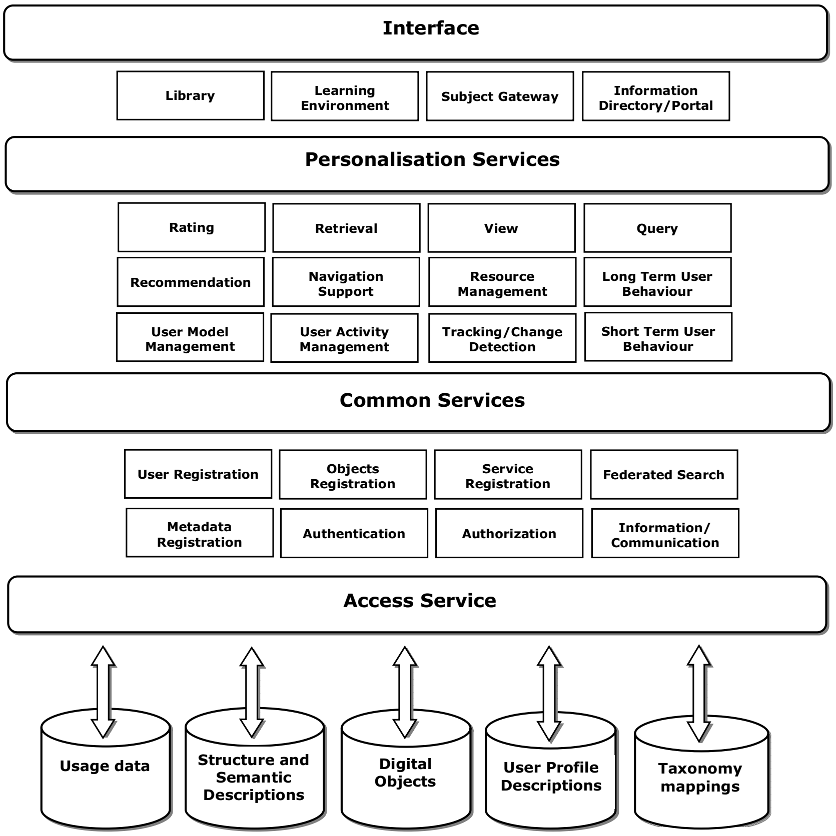

Fig. 1. A set of identified services for personalised access.

The model of Fig. 1 shows a proposed set of fundamental services for structured information spaces. The services are organised into logical groups but no explicit association among service functional definitions is implied. The “Personalisation” group identifies services that can be used to support functionalities for personalisation of the integrated information space (see Table 1). For example, a Retrieval service may provide personalised information seeking by allowing users to browse digital objects or conduct augmented keyword-based searching. This type of functionality may exploit taxonomies descriptions or metadata properties of the objects as well as user profile elements, such as goals, preferences, cognitive style etc. A service that monitors user Long Term Behaviour can collect user's general preferences so that decisions can be made for the presentation, layout and content of the pages he/she visits. This can be used for example to map digital objects and activities against specific competencies, and allow applications, such as a simple portal (see top level of Fig. 1), to automatically configure themselves for particular user(s) as well as to alleviate manual entering of user preferences into multiple application interfaces, such as portal, library, learning environment etc.

Personalisation Service	Expected functionality
Rating	Support for the use of secondary metadata (user ratings and text annotations) for resources.
Retrieval	Browsing though the digital objects based on taxonomies descriptions.
View	Generate personalised views over the digital objects and schemata.
Query	Provide query facilities over structured and semantic descriptions.
Recommendation	Recommend information content based on application-specific user history and behaviour, and metadata descriptions.
Navigation Support	Support navigation though the information space.
Resource Management	Automatically determine information about appropriate search terms and the structure of metadata records that will be returned to them; support retrieval, description, and organisations of resources.
Long Term User Behaviour	Support the mapping of digital objects and activities against specific competencies; persistent between sessions.
User Model Management	Support the management of individual and group user models. Includes policies for updating and registering user models.
User Activity Management	Initialise services with user preferences information.
Tracking/Change Detection	Track/detect changes in the objects descriptions/metadata as well as in the user model specifications; may call other services to translate changes from one schema to another as well as other access services.
Short Term User Behaviour	Collects information about user model attributes that correlate with functions of the application and the behaviour of the user.

Table 1. Overview of personalisation services for information access.

The "Common Services" group identifies services that may be common across different application domains; e.g. a Search service that supports finding information resources either using a simple query grammar or multiple search types (when search results are collected from across multiple types of search then a Federated Search service is used).

3 User aspects and User Data Models for Personalisation

Personalised access to information is offered on the basis of "understanding" the user. Users seeking for information increasingly need support in order to avoid disorientation. Particularly when browsing an information space, users may fail to develop a holistic understanding of how all the information fits together and as a consequence may formulate unsuccessfully their search goals, information needs, and miss locating relevant content [29]. Moreover, as virtual spaces tend to be immense, dynamic, and fragmented, understanding their organization or the organization of the search results may lead to an ongoing learning process for the user.

Although a variety of data models are available for describing user aspects, such as OUNL-EML, PALO, PAPI, IMS-LIP, ARIADNE, there is no standard way to represent application-specific user models on the Web [10, 12]. Nevertheless, the provision of personalisation services requires creating and updating a user model for each user or for each user group, where the dimensions of the different user models may differ in their semantic descriptions. Hence we identify below nine dimensions of a user data model for structured information spaces. Our choices have been informed by suggestions made in [15] and include:

(i) Personal data, such as gender, age, language, culture, affect the perception of the interface layout, and should be taken into account when designing personalisation services. For example, the preferences of males and females differentiate remarkably in terms of navigation support [4], attitudes [11], information seeking strategies [16,31] and media preferences [22].

(ii) Cognitive or learning styles refer to a user's information processing habits and have an impact on user's skills and abilities, such as preferred modes of perceiving and processing information, and problem solving [3, 19]. They can be used to personalise the navigation support, the presentation and organisation of the content and search results. [20].

(iii) Device information concerns the hardware used for access and affects personalisation services in terms of screen layout and bandwidth limitations [5].

(iv) Context-related data capture the physical environment from where the user is accessing the information and can be used to infer the user's goals [18].

(v) User history data capture user past interaction with the system, e.g. visited pages that contain pointers to specific keywords [23], or browsing habits [25], and can be used under the assumption that users' future behaviour will be almost similar to their past behaviours.

(vi) User preferences and interests are usually provided in the form of keywords or topics of interest for that user [23, 27].

(vii) Goal-related data indicate the reason for which that user is searching information for that particular session [14, 24]. For example it is not the same to search information about China as a tourist or as a student writing a school report.

(viii) System experience indicates the knowledge of that particular user about the information space. For example, system experience may depend on users' familiarity with a digital library features and functionalities [26], or with her familiarity with learning environments [21, 30]. It can be used to personalise the navigation, the search results or provide intelligent help.

(ix) Domain expertise relates to the existing level of understanding of a particular user on the domain knowledge. The level of expertise of a user can vary with the domain and influences the navigation behaviour [9,17].

As already mentioned, it is unlikely that all the information required by any particular personalised information space can be captured in the elements of a specific data models. Models, such as PAPI and IMS-LIP include elements for covering dimensions such as user history data and goal-related data but other dimensions of the user model may require mixing, adapting and sometimes extending a data model to meet specific application requirements for the personalisation functionalities [10,12,13]. This has also been considered in the context of semantic web to enable semantically enhanced educational systems to provide personalisation services [7,8]. Encoding user profiles in RDF provides flexibility to include elements from multiple schemata, to enrich them with additional elements, when necessary, and to maintain interoperability with other systems [7,8].

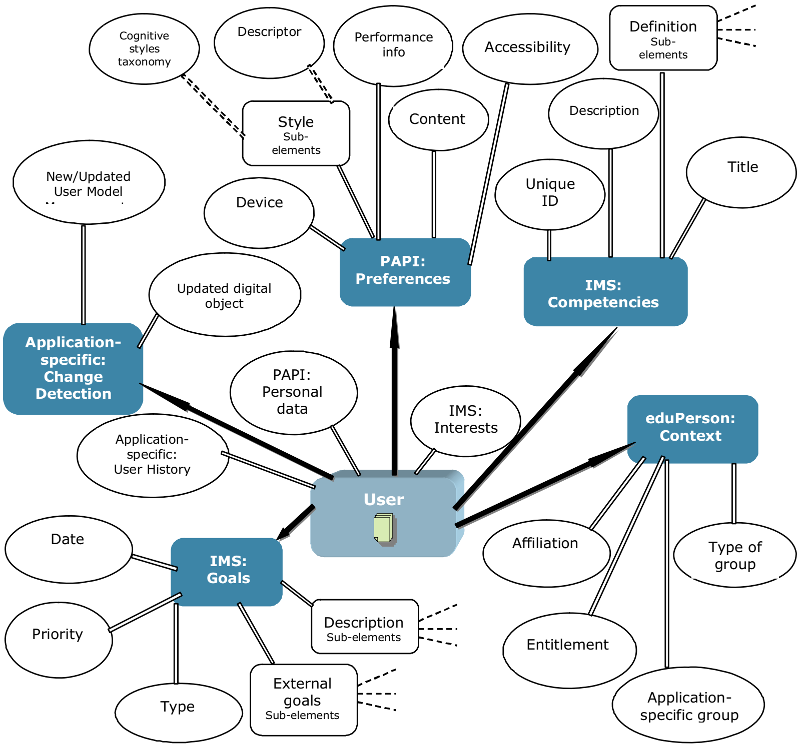

Fig. 2. A fragment of the user model schema.

Fig. 2 adopts this approach providing a simplified view of a user model using an RDF-like encoding. The schema of Fig. 2 uses elements of the PAPI (Personal and Private Information) standard, the IMS (Instructional Management System) metadata specification, as well as application-specific elements, such as the Change Detection element. Also, the Style element, shown in Fig. 2, provides an example of how cognitive styles can be incorporated in the user model schema by extending the PAPI shema (or the *catalog* and *entry* elements of the IMS schema). The style *taxonomy* can include several cognitive style categorisations, such as Witkin; Honey-Mumford and so on; while the *descriptor* element can take values in the set of field dependent/field independent or reflector/theorist/activist/pragmatist respectively. Moreover, the user model may include elements of the eduPerson schema which defines how a subset of the user information might be represented in an enterprise directory. Whilst, the IMS specification defines application independent structured data models for representing various pieces of user information, the eduPerson elements of the user schema allow authorised users and services to access information regardless of where or how the original information is stored. Not all user model elements are necessary in order to implement a given service.

3 Discussion and Future Work

The service-oriented approach models an information space on the basis of services, which work on data structures/objects, and processes that describe sequences of steps and the services and data involved in each step, in order to tailor the information and the interface to the needs of the individual. This is visualised as layered software architecture in Fig. 3, in an attempt to provide an overview of this approach.

One key challenge of this approach is defining what components are needed and how they should be connected so that they have minimum dependencies in order to be recombined for different purposes. (Components can consist of objects, services and processes that are related to each other, e.g. a component can organise the operation of other components.) Another challenge is identifying what services components should offer. (A service can be used to connect one component with another, or as a method applied on an object.)

Personalisation in this context emerges through the aggregation of a set of services that implement a personalised function. It can also be materialised by creating, managing and storing "personal views" or relationships between information from a diverse set of existing applications (see Application delivery layer in Fig. 3). These can be tailored to the needs of individual users by combining components (which will provide the necessary functionality) and assembling services from a set of components to reduce implementation cost. For example, new types of "personal" information spaces can be composed, multiple user interfaces or portals, tailored to specific users or tasks, can be produced in this way (see Application delivery layer in Fig. 3). This of course requires a framework for the user interface, which as shown in Fig. 3 can take different forms to that to manage the communication between layers, support navigation and content presentation to each user. The user interface is supported by Application and Personalisation Services as shown in the Services layer of Fig. 3.

In general, the Services layer creates the mapping between applications, systems and data, and the "service-oriented model". Two levels of services can be envisaged: (i) high-level services include services, processes and objects/data structures that are shared across applications, aggregate low level services functionality, manage user/application data, define processes, control objects/services etc. For example, a personalisation service can translate application functionality into user interface features. This may depend on the state of an application and/or previous user activity. The personalisation service may behave differently when the user is not authenticated (see Common Services in Fig. 1). Certain personalisation services may be offered to the user only when an application is at a particular state or when previously called services have performed certain actions. Low level services, such as an authentication service, can be considered as general purpose; they do not rely on other services, and are standardised across all applications. For example, these may include services for object/data registration, communication (cf. Fig. 1).

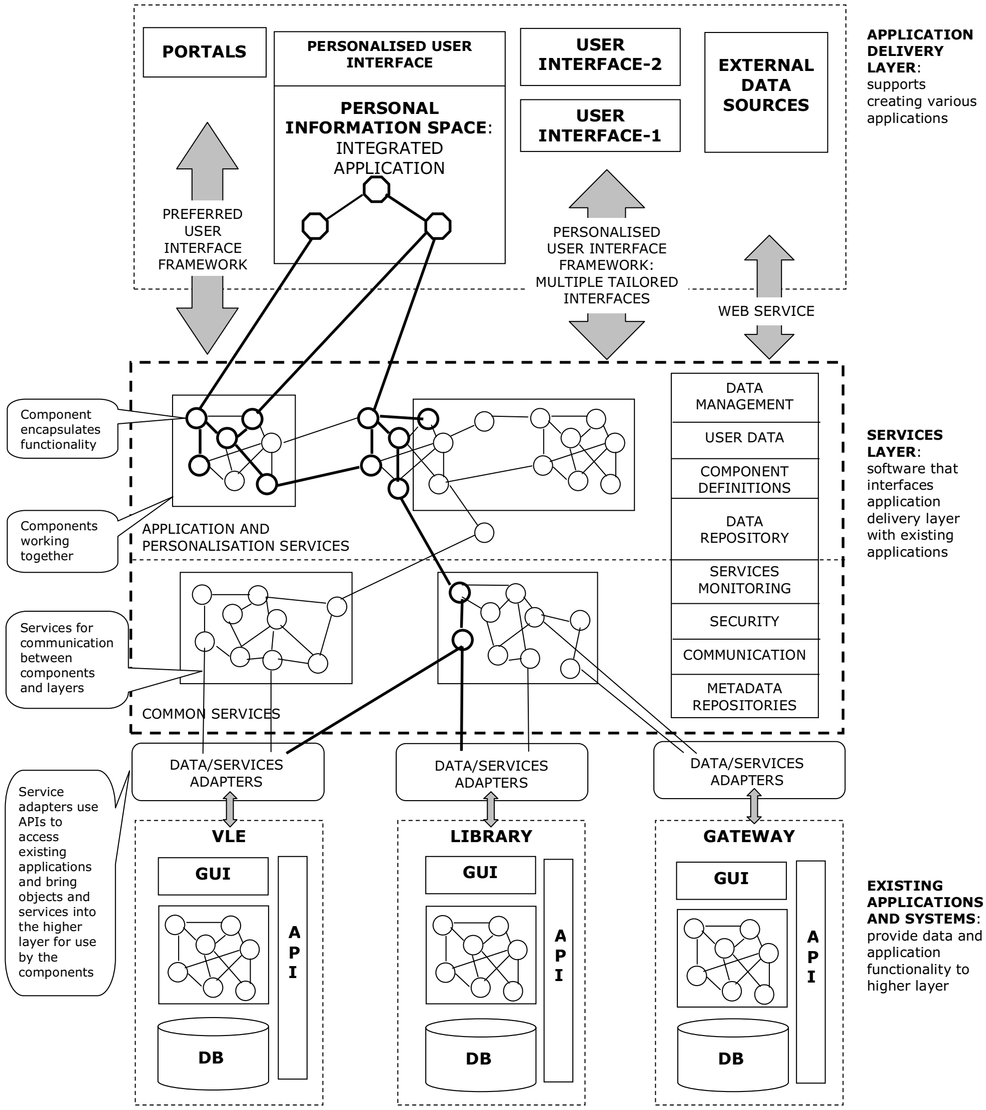

Fig. 3. High level description of generic service-oriented architecture that supports various types of personalisation.

In personalised information spaces this could take different forms: (i) personalisation of content, making possible for each user to create a "personal" information space that contains only the information that is interesting and relevant to that user, (ii) user navigation support though the information space (iii) tailored information retrieval, filtering and recommendation, simplifying the process of locating and filtering the vast amount of information that a user can access. In our future work we are planning to explore in detail several aspects of this approach.

References

1. Ardissono L., Goy A., Petrone G., Segnan M.: Interaction with Web Services in the Adaptive Web, Lecture Notes in Computer Science, 3137. Proc. 3rd Int. Conf. Adaptive Hypermedia and Adaptive Web-Based Systems, Eindhoven, 14-23, 2004.
2. Brusilovsky, P.: Adaptive Educational Hypermedia: From generation to generation. Proceedings of 4th Hellenic Conference on Information and Communication Technologies in Education, Athens, Greece, 19-33, 2004.
3. Chen S., Magoulas G.D. and Macredie R.: Cognitive Styles and Users' Reponses to Structured Information Representation, International Journal of Digital Libraries, 4, 2, 93-107, 2004.
4. Czerwinski, M., Tan, S.D., Robertson, G.G.: Women take wider view. Proceedings of the ACM SIGCHI 2002 Conference, Minneapolis, MN, USA, 195–202, 2002.
5. Cohen, D., Herscovici, M., Petruschka,Y., Maarek, Y.S., Soffer: A. Personalized Pocket Directories for Mobiles Devices. In Proceedings of the 11th ACM Int. Conf. on World Wide Web, 627-638, 2002.
6. Colomb R. M.: Information Spaces: The Architecture of Cyberspace, Springer-Verlag, London, 2002.
7. Dolog P and Henze N.: Personalization Services for Adaptive Educational Hypermedia. In Proc. of International Workshop on Adaptivity and User Modelling in Interactive Systems, LLA, Karlsruhe, Germany, October 2003.
8. Dolog P. and Nejdl W.: Challenges and Benefits of the Semantic Web for User Modelling. In Proc. of AH2003 workshop at 12th World Wide Web Conference, Budapest, Hungary, May 2003.
9. Ford, N., Chen, S.: Individual Differences, Hypermedia Navigation and Learning: An Empirical Study, Journal of Education Multimedia and Hypermedia, 9, 4, 281-311, 2000.
10. Godby, C.J. What Do Application Profiles Reveal about the Learning Object Metadata Standard? Ariadne, 4, 2004. Online at http://www.ariadne.ac.uk/issue41/godby/intro.html.
11. Jacobson, F. F. Gender differences in attitudes toward using computers in libraries: An exploratory study, Library & Information Science Research, 13, 267–279, 1991.
12. Heery, R. and Patel, M. Application profiles: mixing and matching metadata schemas, Ariadne, 25, 2000. Online at http://www.ariadne.ac.uk/issue25/app-profiles/intro.html
13. Keenoy K., Poulovassilis A., Christophides V., Rigaux P., Papamarkos G., Magkanaraki A., Stratakis M., Spyratos N., Wood P.T., Personalisation Services for Self e-Learning Networks, Koch, Fraternali and Wirsing (eds.) Proceedings of the International Conference on Web Engineering 2004, Munich, Germany, Lecture Notes in Computer Science, 3140, 215-219, Springer, 2004.
14. Klaus P. Jantke, Gunter Grieser, and Steffen Lange. Adaptation to the learners' needs and desires by induction and negotiation of hypotheses. In Michael E. Auer and Ursula Auer, editors, International Conference on Interactive Computer Aided Learning, ICL 2004, Sept. 29 - Oct. 1, 2004, Villach, Austria (CD-ROM), 2004.
15. Kobsa, A., Koenemann, J., Pohl, W., Personalised hypermedia presentation techniques for improving online customer relationships, The Knowledge Engineering Review, 16, 2, 111-155, 2001.
16. Large, A., Beheshti, J. and Rahman, T. Gender differences in collaborative Web searching behavior: an elementary school study, Information Processing & Management, 38, 3, 427-443, 2002.
17. Last, D. A., O'Donnell, A. M., and Kelly, A. E. The Effects of Prior Knowledge and Goal Strength on the use of Hypermedia Journal of Educational Multimedia and Hypermedia. 10, 1, 3-25, 2001.

18. Maamar, Z., AlKhatib, G., Mostéfaoui, S. K., Lahkim, M. B., and Mansoor, W. Context-based Personalization of Web Services Composition and Provisioning, Proceedings of 30th Euromicro Conference, August 31-September 03, Rennes, France 396-403, 2004.
19. Magoulas, G., Papanikolau, K., Grigoriadou, M.: Adaptive web-based learning: accomodating individual differences through system's adaptation. In British Journal of Educational Technology, 34, 4, 1-19, 2003.
20. Magoulas, G. D., Chen, S. Y., and Dimakopoulos, D. A Personalised Interface for Web Directories based on Cognitive Styles. In User-Centered Interaction Paradigms for Universal Access in the Information Society: 8th ERCIM Workshop on User Interfaces for All, Vienna, Austria, June 28-29, 2004, Revised Selected Papers. Lecture Notes in Computer Science, vol. 3196, Springer-Verlag, 159-166.
21. Mitchell, T. J. F., Chen, S. Y. and Macredie, R. D. Hypermedia Learning and Prior Knowledge: Domain Expertise vs. System Expertise, Journal of Computer Assisted Learning, 21, 53-64, 2005.
22. Parush, A. and Bermanb, D. Navigation and orientation in 3D user interfaces: the impact of navigation aids and landmarks, International Journal of Human-Computer Studies, 61, 3, 375-395, 2004.
23. Rocha, L.M,: TalkMine: a Soft Computing Approach to Adaptive Knowledge Recommendation. In Soft Computing Agents: New trends for designing autonomous systems, Vincenzo Loia and Salvatore Sessa (Eds.). Series on studies in Fuzziness and soft computing. Physica-Verlag, Springer, 89-116, 2002
24. Sellen, A. J., Murphy, R., Shaw, K. L. How knowledge workers use the web, CHI 2002, Proceedings of the ACM Conference on Human Factors and Computing Systems, Minneapolis, MN, USA, 2002.
25. Sugiyama, K., Hatano, K., Yoshikawa, M., Adaptive web search based on user profile constructed without any effort from users, Proceedings of the 13th Inter Conference on World Wide Web, New York, NY, USA, 675 – 684, 2004.
26. Stelmaszewska, H. Blandford, A., Buchanan G. Designing to change users' information seeking behaviour: a case study. In Adaptable and Adaptive Hypermedia Systems, Chen S., and Magoulas G.D. (Eds.), IRM Press, 1-18, 2005.
27. Tanudjaja, F. and Mui, L. Persona: A Contextualized and Personalized Web Search. In Proc. of the 35 th Annual Hawaii Int. Conf. on System Sciences, 1232 -1240, 2002.
28. Theng, Y.L., Duncker, E., Mohd-Nasir, N.: Design Guidelines and User-Centred Digital Libraries, Proceedings of the Third European Conference on Research and Advanced Technology for Digital Libraries, LNAI 1696, pp. 167-183, 1999.
29. Thong, J.Y.L., Hong, W., and Tam, K.Y.: Understanding user acceptance of digital libraries: What are the roles of interface characteristics, organizational context, and individual differences? International Journal of Human-Computer Studies, 57, 3, 215-242, 2002.
30. Torkzadeh G. and Van Dyke T.P. Effects of training on Internet self-efficacy and computer user attitudes. Computers in Human Behavior 18, 479–494, 2002.
31. Vaughan, L. Q. Analytical searching versus browsing: The effects of gender, search task and search experience, Canadian Journal of Information and Library Science, 18, 1–13, 1993.
32. Xie, X., Zeng, H.J., Ma, W.Y: Enabling personalization services on the edge. ACM Multimedia, 263-266, 2002

WordNet-based User Profiles for Semantic Personalization

G. Semeraro, M. Degemmis, P. Lops, and I. Palmisano

Dipartimento di Informatica
Università di Bari
Via E. Orabona, 4 - 70125 Bari - Italia
{semeraro,degemmis,lops,palmisano}@di.uniba.it

Abstract. Algorithms designed to support users in retrieving relevant information base their relevance computations on user profiles in which representations of the users' interests are maintained. A crucial issue is that users want to retrieve information on the basis of *conceptual content*, but words provide unreliable evidence about the content of documents. This paper explores a possible solution for this kind of problems: the adoption of supervised machine learning techniques to induce *semantic user profiles* from text documents.

1 Introduction

Current search services take a "one fits all" approach, which takes little account of the user's individual needs and preferences. Recent developments at the intersection of information retrieval, information filtering, machine learning, user modeling and natural language processing offer novel solutions for *personalized information access*. Most of this work focuses on the use of machine learning algorithms for the automated induction of a structured model of a user's interests, the *user profile*, from labeled text documents. The keyword approach to searching suffers from problems of POLYSEMY, the presence of multiple meanings for one word, and SYNONYMY, that stands for multiple words having the same meaning. The result is that, due to synonymy, relevant information can be missed if the profile does not contain the exact keywords occurring in the documents and, due to polysemy, wrong documents could be deemed as relevant. These problems call for alternative methods able to learn *semantic* profiles that capture key concepts representing users' interests from relevant documents. Semantic profiles will contain references to concepts defined in lexicons or, in a further step, ontologies. This paper shows how the content-based algorithms for learning user profiles can be extended using WordNet [1] as a reference lexicon in substituting word forms with word meanings into profiles. The paper is organized as follows: after introducing the task of learning user profiles as a text categorization problem in Section 2, in Section 3 we present the relevance feedback approach we adopted to accomplish this task. In Section 4 a strategy to represent documents and user profiles using WordNet synsets is proposed. Section 5 describes the experimental evaluation of semantic user profiles, while some conclusions are drawn in Section 6.

2 Learning User Profiles as a Text Categorization Problem

The content-based paradigm for information filtering (IF) is analog to the relevance feedback in information retrieval literature [2], which adapts the query vector by iteratively absorbing users relevance judgments on newly returned documents. In the IF paradigm, the tuned query vector is a profile model, specifying both keywords and their informative power. A new item relevance is measured by computing a similarity measure between the query vector and the items feature vector. Machine Learning (ML) techniques are used to generate a predictive model that, when given a new information item, will predict whether the new item is likely to be of interest, based on information previously labeled by the user. The ML techniques generally used are those that are well-suited for text categorization (TC) [3]. TC is the task of assigning a Boolean value to each pair $\langle d_j, c_i \rangle \in D \times C$, where D is a domain of documents and $C = \{c_1, \ldots, c_n\}$ is a set of predefined categories. A value of *True* assigned to $\langle d_j, c_i \rangle$ indicates a decision to assign c_i to d_j, while a value of *False* indicates the opposite decision. The task is to approximate the unknown target function $\Phi : D \times C \longrightarrow \{True, False\}$, that describes how documents should be classified, by means of a function $\Phi' : D \times C \longrightarrow \{True, False\}$ called the classifier or the *model* such that Φ and Φ' "coincide as much as possible". In the ML approach to TC, an inductive process automatically builds a text classifier by learning, from a set of *training documents* - documents labeled with the categories they belongs to - the features of the categories. We consider the problem of learning user profiles as a binary TC task: each document has to be classified as interesting or not with respect to the user preferences. Therefore, the set of categories is restricted to c_+, that represents the positive class (user-likes), and c_- the negative one (user-dislikes). We present a relevance feedback method able to learn profiles for content-based filtering. The accuracy of the keyword-based profiles inferred by this method will be compared with advanced semantic user profiles obtained by the same method using an indexing procedure based on WordNet.

2.1 Documents Representation

The representation that dominates the TC literature is known as *bag of words* (BOW). In this approach each feature corresponds to a single word found in the training set. In our application scenario, items to be suggested to users are movies. Each movie is represented by a set of *slots*, where each slot is a textual field corresponding to a specific feature of the movie: *title*, *cast*, *director*, *summary* and *keywords*. The text in each slot is represented using the BOW model taking into account the occurrences of words in the original text. Thus, each instance is represented by five BOWs, one for each slot. This strategy considers separately the occurrences of a word in the slots in which it appears. The idea behind this approach is that by considering the number of occurrences separately in each slot could supply a more effective way to catch the discriminatory power of a word in a document.

2.2 Related Works

Content-based systems have been used successfully in various domains.

Syskill & Webert [4] is an agent that learns a user's interests saved as a user profile used to identify interesting Web pages. The learning process is conducted by using algorithms like Bayesian classifiers, a nearest neighbor algorithm and a decision tree learner. Mooney and Roy [5] adopt a naïve Bayes text classifier in their *LIBRA* system, that makes content-based book recommendations exploiting the product descriptions obtained from the Web pages of the Amazon store. SiteIF [6] is a personal agent that exploits a sense-based representation to build a model of the user's interests as a semantic network whose nodes represent senses (not just words) of the documents requested by the user. Several methods have been proposed for integrating lexical information to training documents for text categorization. A study by Rodriguez et al. [7] used WordNet to enhance neural network learning algorithms. This approach only made use of synonymy and involved a manual word sense disambiguation step, whereas our approach uses synonymy and hypernymy and is completely automatic. Scott and Matwin [8] propose to expand each word in the training set with all the synonyms extracted from WordNet for it, including those available for each sense in order to avoid a word sense disambiguation process. This approach has shown a decrease of effectiveness in the classifier obtained, mostly due to the word ambiguity problem. Some researches have also applied WordNet to information retrieval tasks. In [9], it is proposed a retrieval strategy that adapts a classical vector space based system using synsets as indexing space instead of word forms.

3 A Relevance Feedback Method for User Profiling

In the Rocchio algorithm [10], documents are represented with the vector space representation and the major heuristic component is the TFIDF (Term Frequency/Inverse Document Frequency) word weighting scheme [2]:

$$\text{TFIDF}(t_k, d_j) = \underbrace{\text{TF}(t_k, d_j)}_{\text{TF}} \cdot \underbrace{log \frac{N}{n_i}}_{\text{IDF}} \tag{1}$$

where N is the total number of documents in the training set and n_i is the number of documents in which the term t_k appears. $TF(t_k, d_j)$ is a function that computes the frequency of the token t_k in the document d_j. Learning is achieved by combining document vectors of positive and negative examples into a prototype vector $\overrightarrow{c}$ for each class in the set of classes C. The method computes a classifier $\overrightarrow{c_i} = \langle \omega_{1i}, \ldots, \omega_{|T|i} \rangle$ for category c_i (T is the *vocabulary*, that is the set of distinct terms in the training set) by means of the formula:

$$\omega_{ki} = \beta \cdot \sum_{\{d_j \in POS_i\}} \frac{\omega_{kj}}{|POS_i|} - \gamma \cdot \sum_{\{d_j \in NEG_i\}} \frac{\omega_{kj}}{|NEG_i|} \tag{2}$$

where ω_{kj} is the $TFIDF$ weight of the term t_k in document d_j, POS_i and NEG_i are the set of positive and negative examples in the training set for the specific class, β and γ are control parameters that allow setting the relative importance of *all* positive and negative examples. To assign a class $\widetilde{c}$ to a document d_j, the similarity between each prototype vector $\overrightarrow{c_i}$ and the document vector $\overrightarrow{d_j}$ is computed and $\widetilde{c}$ will be the c_i with the highest value of similarity. We propose a method that manages documents represented using different slots. If m is the index of the slot, a movie is represented by the concatenation of five BOWs:

$$d_j = \langle w^m_{1j}, \ldots, w^m_{|T_m|j} \rangle$$

where $|T_m|$ is the cardinality of the vocabulary for the slot s_m and w^m_{kj} is the weight of the term t_k in slot s_m of the document d_j, computed as:

$$\text{TFIDF}(t_k, d_j, s_m) = \text{TF}(t_k, d_j, s_m) \cdot log \frac{N}{n_{km}} \tag{3}$$

$\text{TF}(t_k, d_j, s_m)$ is the frequency of term t_k in the document d_j in the slot s_m; the inverse document frequency of the term t_k in the slot s_m is computed as the logarithm of the ratio between the total number of documents N and the number of documents containing the term t_k in the slot s_m.

Given a user u and a set of rated movies in a specific category of interest (for example, *Comedy*), the goal is to learn a profile able to recognize movies liked by the user in that category. The learning process consists in inducing one prototype vector for *each slot*: these five vectors will represent the user profile. Each prototype vector of the profile could contribute in a different way to the calculation of the similarity between the vectors representing a movie and the vectors representing the user profile. Another key issue of our algorithm is that it learns two different profiles $\overrightarrow{p_i} = \langle \omega^m_{1i}, \ldots, \omega^m_{|T_m|i} \rangle$, for a user u and a category c_i by taking into account the ratings given by the user on documents in that category. The rating $r_{u,j}$ on the document d_j is a discrete judgment ranging from 1 to 6. It is used to compute the coordinates of the vectors in both the positive and the negative user profile:

$$\omega^m_{ki} = \sum_{\{d_j \in POS_i\}} \frac{\omega^m_{kj} \cdot r'_{u,j}}{|POS_i|} \quad (4) \qquad \omega^m_{ki} = \sum_{\{d_j \in NEG_i\}} \frac{\omega^m_{kj} \cdot r'_{u,j}}{|NEG_i|} \quad (5)$$

where $r'_{u,j}$ is the normalized value of $r_{u,j}$ ranging between 0 and 1 (respectively corresponding to $r_{u,j} = 1$ and 6), $POS_i = \{d_j \in T_r | r_{u,j} > 3\}$, $NEG_i = \{d_j \in T_r | r_{u,j} \leq 3\}$, and ω^m_{kj} is the weight of the term t_k in the document t_j in the slot s_m computed as in equation (3) where the *idf* factor is computed over POS_i or NEG_i depending on the fact that the term t_k is in the slot s_m of a movie rated as positive or negative (if the term is present in both positive and negative movies two different values for it will be computed). Computing two different idf values for a term led us to consider the rarity of a term in positive and negative movies, in an attempt to catch the informative power of a term in recognizing interesting movies. Equations (4) and (5) differ from the classical formula in the fact that the parameters β and γ are substituted by the ratings $r'_{u,j}$ that allow to give a different weight to each document in the training set. As regards the computation of the similarity between a profile $\overrightarrow{p_i}$ and a movie $\overrightarrow{d_j}$, the idea is to compute five partial similarity values between each pair of corresponding vectors in $\overrightarrow{p_i}$ and $\overrightarrow{d_j}$. A weighted average of the five values is computed:

$$sim(\overrightarrow{d_j}, \overrightarrow{p_j}) = \sum_{s=1}^{5} sim(\overrightarrow{d_j^s}, \overrightarrow{p_j^s}) \cdot \alpha_s \tag{6}$$

where α_s reflects the importance of a slot in classifying a movie. In our experiments, we used $\alpha_1 = 0.1$ (title), $\alpha_2 = 0.15$ (director), $\alpha_3 = 0.15$ (cast), $\alpha_4 = 0.25$ (summary) and $\alpha_5(keywords) = 0.35$. The values α_s were decided according to experiments not reported in the paper due to space limitations. We considered different values for each α_s and repeated the experiments reported in section 5 using the selected values. The values reported here are those that gave the best predictive accuracy of the profiles. Since the user profile is composed by both the positive and the negative profiles, we compute two similarity values, one for each profile. The document d_j is considered as interesting only if the similarity value of the positive profile is higher than the similarity of the negative one.

4 Semantic User Profiles

We propose a novel document representation used to build *semantic user profiles* taking into account the senses of the words in the training documents. The task of disambiguation consists in determining which of the senses of an ambiguous word is invoked in a particular use of the word [11]. As for sense repository, we have adopted WordNet (version 1.7.1) [1], a large lexical database for English in which nouns, verbs, adjectives and adverbs are organized into *synsets* (*syn*onym *set*s), each representing one underlying lexical concept. Synsets are linked by different semantic relations (IS-A, PART-OF, etc...) and organized in hierarchies. The main advantage of a synset-based document representation is that synonym words belonging to the same synset can contribute to the user profile definition by referring to the same concept. Moreover, the use of a WSD procedure reduces classification errors due to ambiguous words, and consequently allows a better precision in the user model construction. We have addressed the WSD problem by proposing an algorithm based on semantic similarity between WordNet synsets. The idea behind the algorithm is that semantic similarity between synsets is inversely proportional to the semantic distance between synsets in the WordNet IS-A hierarchy [1]. The path length similarity between synsets is used by the WSD procedure to associate the appropriate synset to a polysemous word, as reported in Algorithm 1. Each document in the collection is mapped into a list of WordNet synsets following these steps:

1. each monosemous word w in a slot of a document d is mapped into the corresponding WordNet synset;

2. for each couple of words $\langle noun, noun \rangle$ or $\langle adjective, noun \rangle$ (for instance, "white house"), a search in WordNet is made in order to verify if at least one synset exists for the bigram $\langle w_1, w_2 \rangle$. In the positive case, Algorithm 1 is applied on the bigram, otherwise it is applied separately on w_1 and w_2, using all words in the slot as the context C of w;
3. each polysemous unigram w is disambiguated by algorithm 1, using all words in the slot as the context C of w.

Algorithm 1 The WordNet-based WSD algorithm

procedure WSD(w, d) ▷ find the appropriate synset of a polysemous word w in the document d; w may be also a bigram
    $C \leftarrow \{w_1, ..., w_n\}$ ▷ C is the context of w and it is defined as the window of all words that surround w with a fixed radius. For example, $C = \{w_1, w_2, w_3, w_4\}$ is a window with radius=2, if the sequence of words $\{w_1, w_2, w, w_3, w_4\}$ appears in d
    $S \leftarrow \{s_1, ...s_k\}$ ▷ S is the set of all candidate synsets for w
    $s \leftarrow null$ ▷ s is the synset to be returned
    $score \leftarrow 0$ ▷ $score$ is a similarity score assigned to s
    $T \leftarrow \emptyset$ ▷ T is the set of all candidate synsets for all words in C
    for $j \leftarrow 1, n$ **do**
        if $POS(w_j) = POS(w)$ **then** ▷ $POS(x)$ is the part-of-speech of x
            $S_j \leftarrow \{s_{j1}, ...s_{jm}\}$ ▷ S_j is the set of m possible senses for w_j
            $T \leftarrow T \cup S_j$
        end if
    end for
    for $i \leftarrow 1, k$ **do**
        for all $s_h \in T$ **do**
            $score_{ih} \leftarrow$ SinSim(s_i, s_h) ▷ computing similarity scores between s_i and every synset $s_h \in T$
            if $score_{ih} \geq score$ **then**
                $score \leftarrow score_{ih}$
                $s \leftarrow s_i$ ▷ s is the synset $s_i \in S$ with the highest similarity score with the synsets in T
            end if
        end for
    end for
    return s
end procedure

function SinSim(a, b) ▷ The similarity of the synsets a and b
    $N_p \leftarrow$the number of nodes in path p from a to b
    $D \leftarrow$maximum depth of the taxonomy ▷ In WordNet 1.7.1 $D = 16$
    $r \leftarrow -log(N_p/2D)$
    return r
end function

Algorithm 1 has been used to represent documents belonging to the EachMovie dataset according to the new model, that we call "bag-of-synsets" (BOS): the final representation of a document consists of a list of WordNet synsets recognized from the words in the document. Each slot of a document is processed separately and the occurrences of the synsets (instead of words) are computed. For example, if the words "artificial" and "intelligence" occur in the same slot of a document, in the corresponding BOW we count one occurrence for each word; in the BOS, we count only one occurrence of the synset "{05766061} ⟨*noun.cognition*⟩ ARTIFICIAL INTELLIGENCE, AI – (THE BRANCH OF COMPUTER SCIENCE THAT DEAL WITH WRITING COMPUTER PROGRAMS THAT CAN SOLVE PROBLEMS CREATIVELY)". A clear advantage of this representation regards synonyms. For example, if the words "processor" and "CPU" appear in the same slot of document, in the corresponding BOW we count *one* occurrence for each word, even if they refer to the same concept; in the BOS, we count *two* occurrences of the synset "{02888449} ⟨*noun.artifact*⟩ CENTRAL PROCESSING UNIT, CPU, C.P.U., CENTRAL PROCESSOR, PROCESSOR, MAINFRAME ". The final goal of our investigation is to compare the results of word-based and synset-based user profiles, then we do not modify the structure of the profiles and the learning mechanisms proposed in section 3. The difference with respect to word-based profiles is that synset unique identifiers are used instead of words.

5 Experimental Sessions

The goal of the experiments was to evaluate if synset-based profiles had a better performance than word-based profiles. The documents in the EachMovie dataset have been disambiguated using Algorithm 1, obtaining a reduction of the number of features ($172,296$ words vs. $107,990$ synsets, the reduction is roughly 38%). This result is mainly due to the fact that, thanks to the WSD algorithm, bigrams are represented using only one synset and synonym words are represented by the same synset.

5.1 The EachMovie Dataset

The experimental work has been carried out on a collection of $1,628$ textual descriptions of movies rated by $72,916$ real users, the EachMovie dataset[1]. The movies are rated on a 6-point scale mapped linearly to the interval [0,1]. The content information for each movie was collected from the Internet Movie Database[2] using a crawler. Appropriate preprocessing operations[3] have been applied to obtain the BOW from the original movie descriptions. Movies are categorized into different genres. For each genre or category, a set of 100 users was randomly selected among users that rated n items, $30 \leq n \leq 100$ in that movie category (only for genre 'animation', the number of users that rated n movies was 33,

[1] http://www.research.compaq.com/SRC/
[2] IMDb, http://www.imdb.com
[3] stopwords elimination and stemming.

due to the low number of movies if that genre). In this way, for each category, a dataset of at least 3000 triples (user,movie,rating) was obtained (at least 990 for 'animation'). Table 1 summarizes the data used for the experiments. The number of movies rated as positive and negative for each genre is balanced in datasets 2, 5, 7, 8 (60-65 % positive, 35-40% negative), while is slightly unbalanced in datasets 1, 9, 10 (70-75 % positive, 25-30% negative), and is strongly unbalanced in datasets 3, 4, 6 (over 75% positive).

Table 1. 10 'Genre' datasets obtained from the original EachMovie dataset.

Id Genre	Genre	Number of Movies rated	% POS	% NEG
1	Action	4,474	72.05	27.95
2	Animation	1,103	56.67	43.33
3	Art_Foreign	4,246	76.21	23.79
4	Classic	5,026	91.73	8.27
5	Comedy	4,714	63.46	36.54
6	Drama	4,880	76.24	23.76
7	Family	3,808	63.71	36.29
8	Horror	3,631	59.89	40.11
9	Romance	3,707	72.97	27.03
10	Thriller	3,709	71.94	28.06
		39,298	71.84	28.16

5.2 Experimental Setup and Results

Classification effectiveness is evaluated by the classical Information Retrieval measures *precision* and *recall*, adapted to the case of text categorization [2]. Also used is *F-measure*, a combination of precision and recall. We adopted the Normalized Distance-based Performance Measure (NDPM) [12] to measure the distance between the ranking imposed on items by the user ratings and the ranking predicted by the Rocchio method, that ranks items according to the similarity to the profile of the class *likes*. Values range from 0 (agreement) to 1 (disagreement). In all the experiments, a movie description d_i is considered as *relevant* by a user if the rating is greater or equal than 3, while the Rocchio method considers an item as relevant if the similarity score for the class *likes* is higher than the one for the class *dislikes*. We executed one experiment for each user in the dataset: the ratings of each specific user and the content of the rated movies have been used for learning the user profile and measuring its predictive accuracy, using the aforementioned measures. Each experiment consisted in:

1. selecting ratings of the user and the content of the movies rated by that user;
2. splitting the selected data into a training set Tr and a test set Ts;
3. using Tr for learning the corresponding user profile;
4. evaluating the predictive accuracy of the induced profile on Ts, using the aforementioned measures.

Table 2. Comparison between the BOW and the BOS approach.

Id Genre	Precision		Recall		F1		NDPM	
	BOW	BOS	BOW	BOS	BOW	BOS	BOW	BOS
1	0.72	0.75	0.82	0.86	0.75	0.79	0.46	0.44
2	0.65	0.64	0.66	0.66	0.64	0.63	0.34	0.38
3	0.77	0.85	0.79	0.86	0.77	0.84	0.46	0.48
4	0.92	0.94	0.94	0.96	0.93	0.94	0.45	0.43
5	0.66	0.69	0.72	0.75	0.67	0.70	0.44	0.46
6	0.78	0.79	0.84	0.87	0.80	0.81	0.45	0.45
7	0.68	0.74	0.75	0.84	0.69	0.77	0.41	0.40
8	0.64	0.69	0.74	0.82	0.67	0.73	0.42	0.44
9	0.73	0.76	0.79	0.81	0.74	0.77	0.48	0.48
10	0.74	0.75	0.85	0.84	0.77	0.78	0.45	0.44
Mean	0.73	0.76	0.78	0.83	0.74	0.78	0.44	0.44

The methodology adopted for obtaining *Tr* and *Ts* was the 10-fold cross validation [13]. The results of the comparison between the profiles obtained from documents represented using the two indexing approaches, namely BOW and BOS, are reported in Table 2. We can notice a slight improvement in precision (+3%). Going in more detail, the BOS model outperforms the BOW model on datasets 3 (+8%), 7 (+6%), 8 (+5%). This could be an indication that the improved results are independent from the distribution of positive and negative examples in the datasets: the number of movies rated as positive and negative is balanced in datasets 8, while is strongly unbalanced in datasets 3 and 7. Similar results have been observed as regards recall and F-measure (+4%). Only on dataset 2 we have not observed any improvement. This is probably due both to the low number of rated movies and to the specific features of the movies (in most cases, stories) that makes difficult the disambiguation. NDPM has not been improved, but it remains acceptable. This measure was adopted in order to compare the ranking imposed by the user ratings and the similarity score for the class c_+ (likes): further investigations will be carried out in order to define a better ranking score for computing NDPM, that takes into account the negative part of the profile as well. A Wilcoxon signed ranked test ($p < 0.05$) has been performed in order to validate the results. We considered each experiment as a single trial for the test. The test confirmed that there is a statistically significant difference in favor of the BOS model with respect to the BOS model as regards precision, recall and F-measure.

6 Conclusions and Future Work

We have presented a system that exploits a relevance feedback learning method to induce semantic user profiles from documents represented using WordNet synsets. Our hypothesis that substituting words with WordNet synsets in the indexing phase produces a more accurate document representation that could be successfully used by learning algorithms to infer more accurate user profiles. This hypothesis is confirmed by the experimental results, since, as expected, a synset-based classification allows to prefer documents with high degree of semantic coherence, which is not guaranteed in case of a word-based classification. As a future work, we will evaluate the effectiveness of the WSD algorithm, by comparing its performance to state-of-the-art systems.

Acknowledgments

This research was partially funded by the European Commission under the 6th Framework Programme IST Integrated Project VIKEF - Virtual Information and Knowledge Environment Framework (Contract no. 507173, Priority 2.3.1.7 Semantic-based Knowledge Systems - `http://www.vikef. net`).

References

1. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press (1998)
2. Salton, G., McGill, M.: Introduction to Modern Information Retrieval. McGraw-Hill, New York (1983)
3. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys **34** (2002)
4. Pazzani, M., Billsus, D.: Learning and revising user profiles: The identification of interesting web sites. Machine Learning **27** (1997) 313–331
5. Mooney, R.J., Roy, L.: Content-based book recommending using learning for text categorization. In: Proceedings of the 5^{th} ACM Conference on Digital Libraries, San Antonio, US, ACM Press, New York, US (2000) 195–204
6. Magnini, B., Strapparava, C.: Improving user modelling with content-based techniques. In: Proc. of 8th International Conference on User Modeling, Springer Verlag (2001) 74–83
7. Rodriguez, M.d.B., Gomez-Hidalgo, J.M., Diaz-Agudo, B.: Using wordnet to complement training information in text categorization. In: 2nd Int. Conf. on Recent Advances in NLP. (1997) 150–157
8. Scott, S., Matwin, S.: Text classification using wordnet hypernyms. In: COLING-ACL Workshop on usage of WordNet for in NLP Systems. (1998) 45–51
9. Gonzalo, J., Verdejo, F., Chugur, I., J., C.: Indexing with wordnet synsets can improve text retrieval. In: COLING-ACL Workshop on usage of WordNet for in NLP Systems. (1998)
10. Rocchio, J.: Relevance feedback information retrieval. In Salton, G., ed.: The SMART retrieval system - experiments in automated document processing, Prentice-Hall, Englewood Cliffs, NJ (1971) 313–323
11. Manning, C.D., Schutze, H.: Foundations of Statistical Natural Language Processing. The MIT Press, Cambridge, US (1984)
12. Yao, Y.Y.: Measuring retrieval effectiveness based on user preference of documents. Journal of the American Society for Information Science **46** (1995) 133–145
13. Mitchell, T.: Machine Learning. McGraw-Hill, New York (1997)

Beyond the Commons: Investigating the Value of Personalizing Web Search

Jaime Teevan[1], Susan T. Dumais[2], and Eric Horvitz[2]

[1] MIT, CSAIL, 32 Vassar St., G472,
Cambridge, MA, 02139 USA
teevan@csail.mit.edu
http://www.csail.mit.edu/~teevan
[2] Microsoft Research, One Redmond Way,
Redmond, WA, 94114, USA
{sdumais,horvitz}@microsoft.com

Abstract. We investigate the diverse goals people have when they issue the same query to a Web search engine, and the ability of current search tools to address such diversity, in order to understand the potential value of personalizing search results. Great variance was found in the results different individuals rated as relevant for the same query—even when those users expressed their underlying informational goal in the same way. The analysis suggests that, while current Web search tools do a good job of retrieving results to satisfy the range of intentions people may associate with a query, they do not do a very good job of discerning an individual's unique search goal. We discuss the implications of this study on the design of search systems and suggest areas for additional research.

1 Introduction

Traditional search engines are designed to return a set of documents that match a query. Studies of search engine quality have tended to be based on the ability of search engines to return the set of results that its users want as a population, as opposed to the results that match each individual's unique search goal. For example, at the DARPA Text REtrieval Conference (TREC), relevant documents to a particular query are identified by an expert judge, based on a detailed description of an information need. Ideally the description is explicit enough and the rater skilled enough that the documents selected as relevant are the same ones that another rater would consider relevant.

However, Web search behavior suggests that providing results to an unambiguous query might not be the most appropriate design target for a search engine. Web queries are very short, and it is unlikely that a two- or three-word query can unambiguously describe a user's informational goal. What one person considers relevant to a query like "jaguar" is not necessarily the same as what someone else considers relevant to the same query. Even a seemingly precise query like "PIA 2005" returns Web pages about the Personal Information Access workshop, the Parachute Industry Association, Professional Insurance Agents, the Pacific Institute of Aromatherapy, etc. Further, if Web searchers are not skilled at stating their goal, even longer descriptions may not reliably disambiguate intent.

We report on a study of the ability of current Web search engines to provide relevant documents to users, in order to understand how future search tools can be built to best meet the needs of their users. Understanding relevance is a complex problem [11, 13], and we address only a small portion of it in our work. Our analysis is aimed at assessing the relationship between the rank of a search result as returned by a Web search engine and the individual's perceived relevancy of the result. We find a considerable mismatch due to a variation in the informational goals of users issuing similar queries. The study suggests personalization of results via re-ranking would provide significant benefit for users. We conclude with a discussion of how the results of this study should triage future research.

2 Methods

We conducted a study in which 15 participants evaluated the top 50 Web search results for approximately 10 queries of their choosing. Participants were employees of a large corporation. Their job functions included administrators, program managers, software engineers and researchers. All were computer literate and familiar with Web search.

Web search results were collected from a "Top Choice" search engine, as listed by Search Engine Watch. For each search result, the participant was asked to determine whether they personally found the result *highly relevant*, *relevant*, or *irrelevant*. So as not to bias the participants, the results were presented in a random order.

The queries evaluated were selected in two different manners, at the participants' discretion. In one approach (*self-selected queries*), users were asked to choose a query to mimic a recently performed search, based on a diary of searches they were asked to keep during the day. Thus, we believe that the self-selected queries closely mirrored the searches that the participants conducted in the real world.

In another approach (*pre-selected queries*), users were asked to select a query from a list of queries that were formulated to be of general interest (e.g., *cancer, Bush, Web search*). Although users did not generate these queries themselves, they were free to choose the pre-selected queries they found most interesting, and thus presumably only chose queries that had some meaning to them. By using pre-selected queries, we were able to explore the consistency with which different individuals evaluated the same results. Such data would have been difficult to collect using only self-selected queries, as it would have required us to wait until different participants coincidentally issued the same query on their own. We validate the conclusions drawn from pre-selected queries with data from the self-selected queries.

For both the self-selected queries and the pre-selected queries, participants were asked to write a more detailed description of the informational goal or *intent* they had in mind when they issued the query. Because the pre-selected queries were given to the user, the user had to create some intent for these queries. However, by allowing them to decide whether or not they wanted to evaluate a particular query, we sought to provide them with a query and associated results that would have some meaning for them.

We collected a total of 137 queries. Of those, 53 were pre-selected queries and 85 were self-selected. The number of users evaluating the same set of results for the pre-selected query ranged from two to nine. Thus we had evaluations by different people for the same queries drawn from the pre-selected set of queries, as well as a number of evaluations for the searches that users had defined themselves.

3 Rank and Rating

We used the data we collected to study how the results that the Web search engine returned matched our participants' search goals. We expected them to match relatively closely, as current search engines seem to be doing well, and in recent years satisfaction with result quality has climbed.

Fig. 1 shows the average result's relevancy score as a function of rank. To compute the relevancy score, the rating *irrelevant* was given a score of 0, *relevant* a score of 1, and *highly relevant* a score of 2. Values were averaged across all queries and all users. Separate curves are shown for the pre-selected (solid line) and self-selected (dashed line) queries. Clearly there is some relationship between rank and relevance. Both curves show higher than average relevance for results ranked at the top of the result list. The correlation between rank and relevance is -0.66. This correlation coefficient is significantly different from 0 ($t(48) = 6.10$, $p < 0.01$). However, the slope of the curves flattens out with increasing rank. When considering only ranks 21-50, the correlation coefficient is -0.07, which is not significantly different from 0. Importantly, there are still many relevant results at ranks 11-50, well beyond what users typically see. This suggests the search result ordering could be improved.

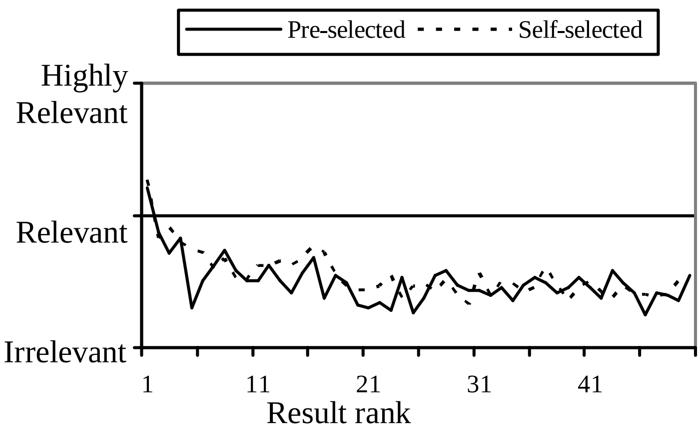

Fig. 1. Average ratings for Web search engine results as a function of rank. There are many relevant results that do not rank in the top ten

The general pattern of results seen in Fig. 1 is not unique to our sample of users or queries. A reanalysis of data from the TREC Web track [4] shows a similar pattern. In the TREC-9 Web track, the top 100 results from 50 Web queries were rated using a similar three-valued scale, *highly relevant*, *relevant* and *not relevant*. Results for one top-performing search systems, uwmt9w10g3, yielded an overall correlation between rank and relevance of -0.81, which drops off substantially to -0.30 for positions 21-50.

4 Same Query, Different Intents

Our analysis shows that rank and rating were not perfectly correlated. While Web search engines do a good job of ranking results to maximize their users' global happiness, they do not do a very good job for specific individuals. If everyone rated the same currently low-ranked documents as highly relevant, effort should be invested in improving the search engine's algorithm to rank those results more highly, thus making everyone happier. However, despite the many commonalities among our participants (*e.g.*, all were employees of the same company, lived in the same area, and had similar computer literacy), our study demonstrated a great deal of variation in their rating of results.

As will be discussed in the following sections, we found that people rated the same results differently because they had different information goals or intentions associated with the same queries. This was evidenced by the variation in the explicit intents our participants wrote for their queries. Even when the intents they wrote were very similar, we observed variation in ratings, suggesting that the participants did not describe their intent to the level of detail required to distinguish their different goals.

4.1 Individuals Rate the Same Results Differently

Participants did not rate the same documents as relevant. The average inter-rater agreement for queries evaluated by more than one participant evaluated was 56%. This disparity in ratings stands in contrast to previous work. Although numbers can't be directly compared, due to variation in the number of possible ratings and the size of the result set evaluated, inter-rater agreement appears to be substantially higher for TREC (*e.g.*, greater than 94% [8]) and previous studies of the Web (*e.g.*, 85% [3]). The differences we observed are likely based in our focus on understanding personal intentions; instead of instructing our participants to select what they thought was "relevant to the query," we asked them to select the results they would want to see personally.

The ratings for some queries agreed more than others, suggesting some queries might be less ambiguous to our population than others. Similarly, some participants gave ratings that were similar to other participants' ratings. It might be possible to cluster individuals, but even the most highly correlated individuals showed significant differences.

4.2 Same Intent, Different Evaluations

We found that our participants sometimes used the same query to mean very different things. For example, the explicit intents we observed for the query *cancer* ranged from "information about cancer treatments" to "information about the astronomical/astrological sign of cancer". This was evident both for the pre-selected, where the user had to come up with an intent based on the query, and self-selected queries, where the query was generated to describe the intent. Although we did not observe any duplicate self-selected queries, many self-selected queries, like "rice" (described as "information about rice university"), and "rancho seco date" (described as "date rancho seco power plant was opened") were clearly ambiguous.

Interestingly, even when our participants expressed the same intent for the same query, they often rated the query results very differently. For example, for the query *Microsoft*, three participants expressed these similar intents:

- "information about microsoft, the company"
- "Things related to the Microsoft corporation"
- "Information on Microsoft Corp"

Despite the similarity of their intent, only one URL (www.microsoft.com) was given the same rating by all three individuals. Thirty-one of the 50 results were rated *relevant* or *highly relevant* by one of these three people, and for only six of those 31 did more than one rating agree. The average inter-rater agreement among these three users with similar intentions was 62%.

This disparity in rating likely arises because of ambiguity; the detailed intents people wrote were not very descriptive. Searches for a simple query term were often elaborated as "information on *query term*" ("UW" → "information about UW", leaving open whether they meant the University of Washington or the University of Wisconsin, or something else entirely). It appears our participants had difficulty stating their intent, not only for the pre-selected queries, where we expected they might have some difficulty creating an intent (mitigated by the fact that they only rated pre-selected queries by choice), but also for the self-selected queries.

Although explicit intents generally did not fully explain the query term, they did provide some additional information. For example, "trailblazer" was expanded to "Information about the Chevrolet TrailBlazer", clarifying the participant was interested in the car, as opposed to, for example, the basketball team. Further study is necessary to determine why people did not include this additional information in their original query, but it does suggest that they could perhaps be encouraged to provide more information about their target when searching. However, even if they did this, they would probably still not be able to construct queries that expressed exactly what wanted. For example, the Trailblazer example above did not clarify exactly what kind of information (*e.g.*, pricing or safety ratings) was sought. This suggests searchers either need help communicating their intent or that search systems should try to infer it.

5 Search Engines are for the Masses

The previous sections showed that our participants ranked things very differently, in ways that did not correspond closely with the Web search engine ranking. We now describe analyses that show that the Web ranking did a better job of satisfying all of our participants than any individual.

5.1 Web Ranking the Best for the Group

In this section, we investigate the best possible ranking we could construct based on the relevance assessments we collected, and compare this ideal ranking with the original Web ranking. For scoring the quality of a ranking, we use *Discounted Cumulative Gain* (DCG), a measure of the quality of a ranked list of results commonly used in information retrieval research [5]. DCG measures the result set quality by counting the number of relevant results returned. It incorporates the idea that highly-ranked documents are worth more than lower-ranked documents by weighting the value of a document's occurrence in the list inversely proportional to its rank (i). DCG also allows us to incorporate the notion of two relevance levels by giving *highly relevant* documents a different gain value than *relevant* documents.

$$\mathrm{DCG}(i) = \begin{cases} G(1) & \text{if } i = 1, \\ \mathrm{DCG}(i-1) + G(i)/\log(i) & \text{otherwise.} \end{cases} \qquad \textbf{(1)}$$

For *relevant* results, we used $G(i) = 1$, and for *highly relevant* results, $G(i)=2$, reflecting their relative importance.

The best possible ranking for a query given the data we collected is the ranking with the highest DCG. For queries where only one participant evaluated the results, this means ranking *highly relevant* documents first, *relevant* documents next, and *irrelevant* documents last. When there are more than one set of ratings for a result list, the best ranking ranks first those results that have the highest collective gain.

We compared how close the best possible rankings were to the rankings the search engine returned. To measure "closeness," we computed the Kendall-Tau distance for partially ordered lists [1]. The Kendall-Tau distance counts the number of pair-wise disagreements between two lists, and normalizes by the maximum possible disagreements. When the Kendall-Tau distance is 0, the two lists are exactly the same, and when it is 1, they are in reverse order. Two random lists have, on average, a distance of 0.5.

We found that for eight of the ten queries where multiple people evaluated the same result set, the Web ranking was more similar to best possible ranking for the group than it was, on average, to the best possible ranking for each individual. The average individual's best ranking was slightly closer to the Web ranking than random (0.5), with a distance of 0.469. The average group ranking was significantly closer ($t(9) = 2.14$, $p < 0.05$) to the Web ranking, with a distance of 0.440. The Web rankings seem to satisfy the group better than they do the individual.

5.2 Gains of Personalization via Re-ranking

Again taking DCG as an approximation of user satisfaction, we found a sizeable difference between our participants' satisfaction when given exactly what they wanted rather than the best group ranking for that query. On average, the best group ranking yielded a 23% improvement in DCG over what the current Web ranking, while the best individual ranking led to a 38% improvement.

The graph depicted in Fig. 2 shows the average DCG for group (dashed line) or personalized (solid line) rankings. These data were derived from the five pre-selected queries for which we collected six or more individual evaluations of the results, although the pattern held for other sets of queries. To compute the values shown, for each query we first randomly selected one person and found the DCG for that individual's best ranking. We then continued to add the additional people, at each step recomputing the DCG for each individual's best rankings and for the best group ranking. As can be seen in Fig. 2, as additional people were added to the analysis, the gap between user satisfaction with the individualized rankings and the group ranking grew. Our sample is small, and it is likely that the best group ranking for a larger sample of users would result in even lower DCG values.

These analyses underscore the promise of providing users with better search result quality by personalizing results. Improving core search algorithms has been difficult, with research leading typically to very small improvements. We have learned that, rather than improving the results to a particular query, we can obtain significant boosts by working to improve results to match the intentions behind it.

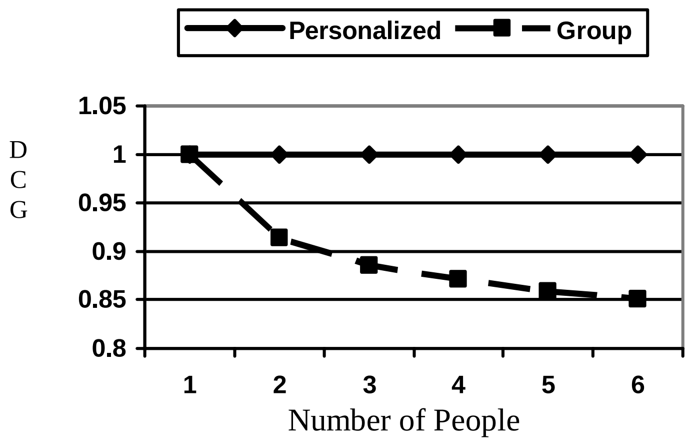

Fig. 2. As more people are taken into account, the average DCG for each individual drops for the ideal group ranking, but remains constant for the ideal personalized ranking

6 Directions in Personalized Search

We believe that Web search tools could be enhanced significantly by considering the variation in relevancy of results for users. We shall now touch on several directions for doing such personalization suggested by the above analysis.

We observed that our participants rated the results to the same queries differently because they had different intents. One solution to ambiguity is to aid users in better specifying their interests and intents. As an example, Google Personal [4] asks users to build a profile of themselves by specifying their interests. Other search systems have tried to help users better express their informational goals through techniques such as relevance feedback or query expansion. While it appears people can learn to use these techniques [2, 8], in practice, on the Web they do not appear to improve overall success [2, 3], and such features have been found to be used rarely. We agree with Nielsen [10], who cites the importance of not putting extra work on the users for personalization. Also, even with additional work, it is not clear that users can be sufficiently expressive. Participants in our study had trouble fully expressing their intent even when asked explicitly to elaborate on their query. In related work, people were found to prefer long search paths to expending the effort to fully specify their query [11].

We believe that another promising approach to personalizing search is to infer users' information goals automatically. Kelly and Teevan [7] give an overview of research done in information retrieval on how implicit measures can be used to help search, highlighting prior contributions focused on helping to improve results for individuals, versus for the general population. In a related paper [15], we describe a search personalization prototype that we have developed which builds on the lessons learned from the study described in this paper. The prototype, named *PS*, uses a person's prior interactions with a wide variety of content to personalize that person's current Web search in an automated manner.

Our study suggests that the results returned by Web search engines represent a range of intentions that people associate with queries. Thus, we believe that personalized search systems could take current Web search results as a starting point for user-centric refinement via re-ranking (e.g., [9, 15]). The original ranking of results by a Web search engine is a useful source of information for a more personalized ranking, and, as we discovered, the first several results are particularly likely to be relevant.

We found that not all queries should be handled in the same manner. For example, we observed that some queries appeared less ambiguous than others and showed less variation among individuals. For such queries, the group ranking (i.e., the current Web search ranking) might be sufficient. A search system that allows users to control how much personalization they receive would improve search relevance while following Neilson's [10] suggestion that users be given control of their content instead of having personalization imposed on them.

7 Conclusion

We have found that there is promise in building tools that perform personalization via re-ranking the results currently provided by current search engines. We have not discussed specific methods to automatically identify users' intentions. Instead we have worked to characterize the range of informational goals associated with queries, and investigated the potential value that can be seen by users via methods that re-rank the list of results provided by search engines.

References

1. Adler, L. M.: A modification of Kendall's tau for the case of arbitrary ties in both rankings. Journal of the American Statistical Society, Vol. 52 (1957) 33–35
2. Anick, P.: Using terminological feedback for Web search refinement: a log-based study. In Proceedings of WWW '04 (2004) 88–95
3. Eastman, C. M. and Jansen, B. J.: Coverage, relevance and ranking: The impact of query operators on Web search engine results. TOIS, Vol. 21(4) (2003) 383–411
4. Google Personal: http://labs.google.com/personalized
5. Hawking, D.: Overview of the TREC-9 Web track. In Proceedings of TREC '00 (2000) 87–102
6. Järvelin, K. and Kekäläinen, J.: IR evaluation methods for retrieving highly relevant documents. In Proceedings of SIGIR '00 (2000) 41–48
7. Jeh, G. and Widom, J.: Scaling personalized Web search. In Proceedings of WWW '03 (2003) 271–279
8. Kelly, D. and Teevan, J.: Implicit feedback for inferring user preference: A bibliography. SIGIR Forum, Vol. 37(2) (2003) 18–28
9. Koenmann, J. and Belkin, N.: A case for interaction: A study of interactive information retrieval behavior and effectiveness. In Proceedings of CHI '96 (1996) 205–212
10. Kritikopoulos, A. and Sideri, M.: The Compass Filter: Search engine result personalization using Web communities. In Proceedings of ITWP '03 (2003)
11. Mizzaro, S.: Relevance: The whole history. JASIST, Vol. 48(9) (1997) 810–832
12. Nielsen, J.: Personalization is overrated. In Jakob Nielsen's Alertbox for October 4 (1998) http://www.useit.com/alertbox/981004.html
13. Schamber, L.: Relevance and information behavior. ARIST, Vol. 29 (1994) 3–48
14. Teevan, J., Alvarado, C., Ackerman, M. S. and Karger, D. R.: The perfect search engine is not enough: A study of orienteering behavior in directed search. In Proceedings of CHI '04 (2004) 415–422
15. Teevan, J., Dumais, S.T. and Horvitz, E.: Personalizing search via automated analysis of interests and activities. To appear in Proceedings of SIGIR '05 (2005)