WEB INTERFACE FOR SEMANTICALLY ENABLED EXPERTS FINDING SYSTEM

Abramowicz Witold, Bukowska Elżbieta, Dzikowski Jakub, Filipowska Agata and Kaczmarek Monika
Department of Information Systems, Faculty of Informatics and Electronic Commerce, Poznan University of Economics
al. Niepodleglosci 10, 61-875 Poznan, Poland

Keywords: Expert finding system, Semantic query interface design, Web interface evaluation.

Abstract: In this paper, we address the issue of designing an interface for a semantic-based expert finding system that allows for creation of sophisticated queries in a user-friendly manner. The designed interface supports the structured way of building queries, the usage of semantic concepts, as well as facilitates the process of defining complex logical structures. We also discuss the procedure and the outcomes of the interface usability evaluation.

1 INTRODUCTION

Today, organizations often take advantage of data available on the Internet to locate experts they require. As the data available is dispersed and of distributed nature, a need appears to support the human resources management process using IT-based solutions, e.g., information extraction and retrieval systems, especially expert finding systems. There are many research and commercial initiatives aiming at development of expert retrieval systems. One of such initiatives is the on-going Polish project eXtraSpec[1]. Its main goal is to combine company's internal electronic documents and information sources available on the Internet to provide an effective way of searching experts with competencies in the given field.

In order to answer users' queries on experts, the eXtraSpec system acquires and extracts information from various sources, and finally, taking advantage of the semantics, reasons over person's characteristics. One of the problems that arises in this context is the simplicity and intuitive use of the interface to formulate the queries by a user. The interface that would allow to formulate complex queries would not fulfil its goal, if users would not be able to use it in practice. Therefore, within the project our aim was to find a balance between the possibilities offered by the querying approach followed within the system and intuitiveness of using the querying interface.

The main goal of this paper is to present the approach followed within the eXtraSpec project, which led to the development of a semantic-based mechanism to retrieve experts. In addition, the evaluation results confirming the usefulness of a system interface being a front-end to this mechanism are presented.

In order to fulfil the mentioned goals, the paper is structured as follows. First, the related work in the area of expert finding systems and interface design is discussed. Next, the eXtraSpec system along with query strategies is presented. Then, remarks on the Web interface developed follow. Finally, the interface evaluation outcomes are presented and discussed. The paper concludes with final remarks.

2 RELATED WORK

2.1 Expert Finding Systems

First systems focusing on expertise identification relied on a database like structure containing a description of experts' skills (e.g., (Yimam-Seid and Kobsa, 2003)). Such systems faced many problems, e.g., how to ensure precise results (Kautz et al., 1996) or how to guarantee the accuracy and validity of stored information. To address these problems, other systems were proposed (e.g., (Campbell et al., 2003), (Hawking, 2004)).

Currently, the Web offers many possibilities to find information on experts. There are a number of contact management or social portals, where users

[1] http://extraspec.kie.ue.poznan.pl/

Witold A., Elżbieta B., Jakub D., Agata F. and Monika K..
WEB INTERFACE FOR SEMANTICALLY ENABLED EXPERTS FINDING SYSTEM.
DOI: 10.5220/0003503902910296
In *Proceedings of the 13th International Conference on Enterprise Information Systems* (ICEIS-2011), pages 291-296
ISBN: 978-989-8425-56-0

can look for experts, potential employees or publish their curricula in order to be found by future employers[2].

Regarding the algorithms applied, at first, standard IR techniques to locate an expert on a given topic were used (Ackerman et al., 2002; Krulwich and Burkey, 1996). Then, to deal with the well-known IR problems, methods such as probabilistic techniques or language analysis techniques to improve the quality of finding systems had been proposed (e.g., (Balog et al., 2006; Fang and Zhai, 2007; Petkova and Croft, 2006; Serdyukov and Hiemstra, 2008)). Finally, the Semantic Web technology has been used to enrich descriptions within expert finding systems e.g., (Dorn et al., 2007).

The expert finding systems have interfaces similar to the regular search engines. A typical interface consist of a field, where a user poses a query and a button that starts searching. A more complicated view is associated with the advanced search, however, even then interface designers work on making it intuitive. Nevertheless, even such simple interfaces may pose severe problems when it comes to keyword specification (Muramatsu and Pratt, 2001).

(Shneiderman et al., 1997) specified eight design guidelines for the development of search user interfaces: offering informative feedback, provision of mechanisms for ordering of results and reformulating queries, showing relevant information, providing alternative interface mechanisms, offering simple error handling mechanisms, striving for consistency in the interface, permitting easy reversal of actions, applying graphic design principles established in the HCI discipline.

These guidelines were taken into account while developing the interface for the eXtraSpec system.

2.2 A Role of an Interface in the Software Engineering Lifecycle

A user interface is often one of the most critical factors for the success or failure of a computerized system (Vliet, 2008). A user judges the quality of a system based on the interface and the way it helps to accomplish users' tasks. In a technical sense, we perceive a user interface as an architecture layer separated from the application logic.

According to (Mayhew, 1999; Vliet, 2008), a well-designed user interface contributes to the quality of a system in the following ways: increased efficiency, improved productivity, reduced errors, reduced training and improved acceptance.

[2]E.g., www.123people.com, sig.ma, www.bizwiz.com, www.xing.com or linkedin.com

Evaluation of user interfaces is often performed using questionnaires (Perlman, 1985; Hornbaek, 2006). There is a number of different approaches to design questionnaires assessing various aspects of usability, validity, reliability e.g. (Norman, 1997; Lewis, 1995; Wiklund, 1994; Lin et al., 1997; Lund, 2001).

The above approaches defined a number of different interface evaluation criteria. These criteria may be further grouped into categories that include inter alia:

- learnability: clarity of wording, support materials, contextual support, support, etc.
- user satisfaction: understandability, ease of learning, etc.
- ease of use: focusing on user friendliness, flexibility, etc.
- usefulness supporting efficiency, productivity, time savings, etc.
- computation efficiency.

A typical questionnaire besides including questions from these categories, usually finishes with an overall judgement of the interface from the user's point of view. This judgement depends however on user's knowledge and his previous experience.

Other type of interface evaluation methods include interviews or user observation methods (Dumas and Redish, 1993).

3 EXTRASPEC SYSTEM

In order to identify the requirements towards the eXtraSpec system, first some searching scenarios a person looking for experts may be interested in, were considered. This allowed especially to specify requirements towards the developed ontology, reasoning mechanism and GUI.

3.1 Querying Strategies

The mentioned scenarios have been specified based on the carefully conducted studies of the literature and interviews with employers. The six most common searching goals are as follows:

1. To find an expert with some experience on a position of interest.

   The requirement on the GUI includes enabling specification of job name (a position of interest) and length of experience required.

2. To find an expert having some specific language skills on a desired level.

Regarding requirements for GUI, there is a need to point to a language of interest (a list), indicate the proficiency level and a certificate name (if desired).

3. To find an expert having some competencies.

   GUI has to give possibility to point to a name of a skill/competency of interest.

4. To find students who graduated recently/will graduate in a given domain.

   A requirement for GUI is to point to a category of educational organization or a specific organization, to name the result, a start and an end date of the education.

5. To find a person having expertise in a specific domain.

   The requirement on GUI is to specify a domain of interest.

6. To find a person with a specific education, competency, job, etc.

   The requirement on GUI is to give possibility to combine various categories into one complex query and make it as easy as possible to specify various logically connected constraints.

The above querying strategies imposed some requirements on the information that should be available for experts as well as ontologies that needed to be developed for the project needs. Requirements on ontologies were further discussed in (Abramowicz et al., 2011). In addition, the following requirements may be defined towards the querying mechanism of the system:

- REQ QM.1: The querying mechanism (QM) MUST allow to build queries in a structured way (i.e., feature: desired value);
- REQ QM.2: The QM MUST support definition of desired values of attributes in a way suitable to the type of data stored within the given feature (i.e., text fields using wild-cards, date fields – after of before certain dates; numbers – less than . . .);
- REQ QM.3: The QM MUST allow to join a subset of selected criteria within the same category into one complex requirement (e.g., category: education; education level: university AND finished date: after 2010 year) using different logical operators;
- REQ QM.4: The QM MUST allow to formulate a set of complex requirements within one category with different logical operators;
- REQ QM.5: It MUST allow to join complex requirements formulated in various profile categories into one criteria with different logical operators;

The logical operators between different set of criteria and criteria themselves, include such operators as: must, should, must not.

The requirements for the system Web interface that result from the above defined requirements towards the entire querying mechanism are detailed and presented in section 4.1.

4 EXTRASPEC WEB INTERFACE DESIGN

The front-end to the eXtraSpec system should enable users to build complex queries describing characteristics of the desired experts. Below we present the identified requirements together with the interface model and implementation-related issues.

4.1 Requirements

During the analysis phase the following requirements for the considered expert finding system interface have been defined:

- REQ IN.1: The interface MUST enable a user to specify constraints on expert's attributes and select whether the value of an attribute is required, desired (but not required) or not allowed.
- REQ IN.2: The interface SHOULD enable grouping of constraints e.g. it should be possible to specify a graduated school and graduation date as one criterion.
- REQ IN.3: The interface SHOULD provide a possibility to build queries which include complementary and alternative constraints.
- REQ IN.4: The interface SHOULD enable providing some of criteria values typed-in as free text (with wildcards) and some of them to be selected from the eXtraSpec system knowledge base.
- REQ IN.5: The interface SHOULD be loosely coupled with the system (following the Seeheim approach).
- REQ IN.6: The interface SHOULD be understandable and easy to use.

The complexity of the querying eXtraSpec system should not affect the interface usability. An average computer-skilled user should facilely express his or her information needs regardless of their complexity.

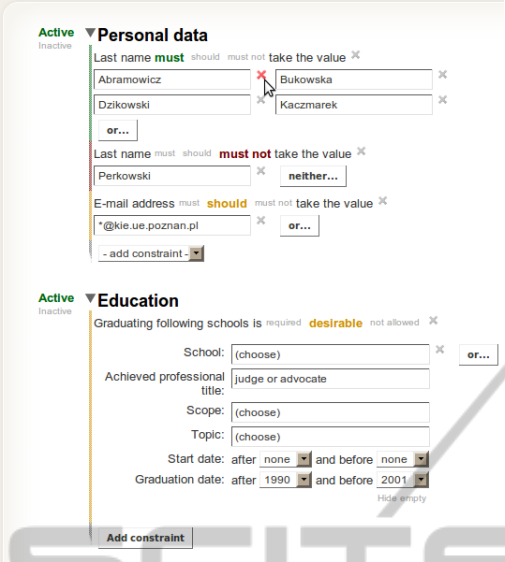

Figure 1: User interface — specifying criteria.

4.2 Conceptual Model of Interface

The search criteria are divided into the following categories: personal data, education, professional experience, foreign languages, courses, certificates, additional skills, organization membership and interests. Most of them reflect categories from the expert's profile in the system, but some (like foreign languages) are created to ease access to frequently used criteria (foreign languages can be found also in the additional skills category).

The categories consist of groups of fields. For example, in the expert education category there is an education group which includes such fields like educational organization, graduation date or achieved professional title. The desired values of these fields are specified in the interface by criteria values, and field groups by criteria groups. Each criterion has a label and a value typed by the user, selected from list or from values tree.

4.3 Definition of Search Criteria

The search criteria are assigned to categories and initially categories are presented to a user. After clicking an appropriate button, a category can be activated or deactivated. Besides, a user can hide criteria in a category (without deactivating it), what should be useful when building complex queries.

A user can also add a group of criteria by clicking a button "Add criteria" or selecting a group name from the list, which is presented instead of a button, when in a category there are multiple groups of criteria. In figure 1 there is group for education which consists of five criteria (school, achieved professional title, scope, topic, start date and graduation date) and there are single-criterion groups, like last name or e-mail address. Within a group, a user can specify alternate criteria, for example in figure 1 alternate last names are specified. The user can add these criteria by clicking a button "or..." or "neither..." (depending on a type constraint for the group). Clicking "x" button will cause removal of a particular criterion. A selected type constraint is related with an appropriate highlighting style of the criteria group.

A user can type some of the criteria, for example last name or e-mail address; with the use of wildcards, when needed. The values for some criteria depend on knowledge base of the eXtraSpec system and possible values are loaded from the ontology. In this case, a user specifies them by clicking on a particular field and then selecting appropriate values from a tree in a pop-up window (see figure 2). The selected items are presented on the bottom of the pop-up window and, finally, in the appropriate field.

5 WEB INTERFACE EVALUATION

Within our research we followed the remote testing approach, i.e., the users were not directly observed while testing the application. The testing procedure was discussed with them during a meeting, and they were also informed what is the aim of the experiment and what is to be tested. Then, they were given a testing task together with an accompanying questionnaire to fill in.

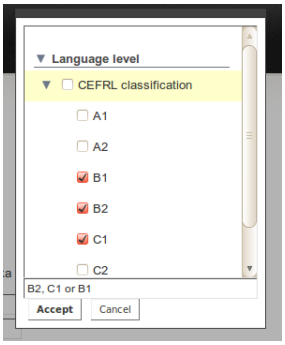

Figure 2: User interface — window for criteria selection.

5.1 Experiment Description

The experiment that aimed at the evaluation of the user interface was carried out in two distinct phases with a 2-weeks' break in-between.

The first phase was focused on learning how to use the interface and pose the queries using the system. The users were provided with a set of free-text queries (considering query strategies presented in section 3.1) and asked to express these queries using the provided interface. These queries reflected common information needs that may occur in experts finding systems. Afterwards, they were to provide the query generated by the system, which was displayed to them in addition to the results. They were also to describe their overall impression from using the system. This way of interviewing users made them focus on providing the query as close to requirements as possible. The user also knew that they may influence the final version of the interface, so they provided also recommendation what should be changed within the system.

In the second phase users again were asked to experiment with the system, however, this time they were asked to provide detailed marks for each feature of the interface. The aim of this phase of the experiment was to evaluate the interface quality.

The users were not initially informed that this is a two-phase experiment.

In both phases the same group of user's participated. The group consisted of 31 students in their final year of the bachelor studies. The students have just finished their software engineering course during which they discussed various examples of user interfaces as well as guidelines for the interface design. Some of the students are also part-time employees of IT companies.

5.2 Experiment Results

As a result of the first phase of experiment 70 query instances with average accuracy at 83% were gathered (all generated queries were checked for completeness). The best reached accuracy is 95% with 12 query instances, the worst 56% with 8 instances.

The first phase of the experiment showed that the current interface makes the system easy to use, but the level of error tolerance or user assistance while generating queries should be improved. The users suggested to include an automcompletion mechanism while writing or some pre-check of data introduced in a field. This however, could lead to limiting the user in expressing the query. In case there are no expert profiles fulfilling the requirements, nothing is returned.

Table 1: Detailed results of experiments.

Category	1	2	3	4	5	Average rating
General	2%	6.5%	15%	59%	18%	3.9
Screen	2%	16%	16%	40%	27%	3.7
Capabilities	0%	2%	13%	32%	53%	4.4
Terminology	0%	16%	18%	42%	24%	3.7
Ease of use	3%	19%	32%	39%	8%	3.3

The detailed evaluation of the system however, was provided in the second phase of the experiment held. Here users, were to answer 12 detailed questions about the features of the system. They were assessing the interface following the Likert's approach (with the use of 1–5 scale) in the below categories:

- general: general impression, appriopriateness for users with various computer abilities;
- screen: distribution of element of the screen, intuitiveness, similarity to front-ends of other known finding systems;
- capabilities: speed and reliability;
- terminology: appropriate labels, icons and messages;
- ease of use.

The best rating the application's front-end gained in the capabilities category and the worst in the ease of use category (mainly because of not enough support offered to the user while defining the query). Results of the evaluation are presented in the table 1.

The most interesting, however, are not the average evaluation results for each category, but their distribution when it comes to the different values of the scale. The system was judged as good or very good by most of the users. They also underlined that the system is suitable also for a user without the expertise in IT (especially when it comes to the advanced querying).

6 CONCLUSIONS

Within this paper we have described the eXtraSpec system together with its querying interface allowing to find persons with specific characteristics. We argue that designing a Web interface for a semantically-enabled system while providing new querying possibilities, poses also new challenges within the interface design process. The conducted evaluation has proven that extending the query definition process using an appropriate form that need to be filled in before submitting the query, while allowing for expressing more complex queries, is still understandable to users, and quality of results achieved re-compensates the effort

needed to be invested into the query definition process.

ACKNOWLEDGEMENTS

The work published in this article was supported by the project titled: "Advanced data extraction methods for the needs of expert search" financed under the Innovative Economy Framework and partially supported by European Union in the European Regional Development Fund programme (contract no. UDA-POIG.01.03.01-30-150/08-01).

REFERENCES

Abramowicz, W., Bukowska, E., Kaczmarek, M., and Starzecka, M. (2011). Semantic-enabled efficient and scalable retrieval of experts. In *International Conference on Information, Process, and Knowledge Management, eKNOW 2011*.

Ackerman, M., Wulf, V., and Pipek, V. (2002). *Sharing Expertise: Beyond Knowledge Management*. MIT Press.

Balog, K., L., A., and De. Rijke, M. (2006). Formal models for expert finding in enterprise corpora. In *Proceedings of the ACM SIGIR*, pages 43–50.

Campbell, C. S., Maglio, P. P., Cozzi, A., and Dom, B. (2003). Expertise identification using email communications. In *CIKM '03: Proceedings of the twelfth international conference on Information and knowledge management*, pages 528–321. ACM Press.

Dorn, J., Naz, T., and Pichlmair, M. (2007). Ontology development for human resource management. In *Proceedings of 4th International Conference on Knowledge Management*, Series on Information and Knowledge Management, pages 109–120.

Dumas, J. and Redish, J. (1993). *A Practical Guide to Usability Testing*. Norwood, NJ: Ablex.

Fang, H. and Zhai, C. (2007). Probabilistic models for expert finding. In *Proceedingsof the ECIR*, pages 418–430.

Hawking, D. (2004). Challenges in enterprise search. In *Proceedings of the 15th Australasian database conference – Volume 27*, ADC '04, pages 15–24, Darlinghurst, Australia, Australia. Australian Computer Society, Inc.

Hornbaek, K. (2006). Current practice in measuring usability: Challenges to usability studies and research. *Int. J. Hum.-Comput. Stud.*, 64:79–102.

Kautz, H., Selman, B., and Milewski, A. (1996). Agent amplified communication. In *Proceedings of the Thirteenth National Conference on Artificial Intelligence (AAAI-96)*, pages 3–9.

Krulwich, B. and Burkey, C. (1996). Contactfinder agent: answering bulletin board questions with referrals. In *Proceedings of the National Conference on Artificial Intelligence*, pages 10–15.

Lewis, J. R. (1995). Ibm computer usability satisfaction questionnaires: Psychometric evaluation and instructions for use. *International Journal of Human-Computer Interaction*, 7(1):57–78.

Lin, H. X., Choong, Y.-Y., and Salvendy, G. (1997). Proposed index of usability: A method for comparing the relative usability of different software systems usability evaluation methods. *Behaviour and Information Technology*, 16 (4/5):267–278.

Lund, A. M. (2001). Measuring usability with the use questionnaire. *Usability Interface*, 8(2).

Mayhew, D. J. (1999). The usability engineering lifecycle. In *CHI '99 extended abstracts on Human factors in computing systems*, CHI '99, pages 147–148, New York, NY, USA. ACM.

Muramatsu, J. and Pratt, W. (2001). Transparent queries: investigation users' mental models of search engines. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 217–224. ACM.

Norman, B. H. . L. S. . K. (1997). Questionnaire administration via the www: A validation and reliability study for a user satisfaction questionnaire. In *Proceesings of the WebNet 97, Association for the Advancement of Computing in Education, Toronto, Canada.*

Perlman, G. (1985). Behavior research methods. *Instruments, and Computers*, 17:203–205.

Petkova, D. and Croft, W. (2006). Hierarchical language models for expert finding in enterprise corpora. In *Proceedings of the 18th IEEE International Conference on Tools with Artificial Intel-ligence*, pages 599–608.

Serdyukov, P. and Hiemstra, D. (2008). Modeling documents as mixtures of persons for expert finding. In *Proceedings of the ECIR*, pages 309–320.

Shneiderman, B., Byrd, D., and Croft, W. (1997). Clarifying search: A user-interface framework for text searches. *D-lib magazine*, 3(1):18–20.

Vliet, H. V. (2008). *Software Engineering*. John Wiley & Sons, Ltd.

Wiklund, M., editor (1994). *Usability in Practice: How Companies Develop User-Friendly Products*. Boston: Academic Press.

Yimam-Seid, D. and Kobsa, A. (2003). Expert finding systems for organizations: Problem and domain analysis and the demoir approach. In *Journal of Organizational Computing and Electronic Commerce*, 13(1), pages 1–24.