Evaluating Semantic Search Systems to Identify Future Directions of Research*

Khadija Elbedweihy[1], Stuart N. Wrigley[1], Fabio Ciravegna[1],
Dorothee Reinhard[2], and Abraham Bernstein[2]

[1] University of Sheffield, Regent Court, 211 Portobello, Sheffield, UK
{k.elbedweihy, s.wrigley, f.ciravegna}@dcs.shef.ac.uk
[2] University of Zürich, Binzmühlestrasse 14, CH-8050 Zürich, Switzerland
{dreinhard, bernstein}@ifi.uzh.ch

Abstract. Recent work on searching the Semantic Web has yielded a wide range of approaches with respect to the style of input, the underlying search mechanisms and the manner in which results are presented. Each approach has an impact upon the quality of the information retrieved and the user's experience of the search process. This highlights the need for formalised and consistent evaluation to benchmark the coverage, applicability and usability of existing tools and provide indications of future directions for advancement of the state-of-the-art. In this paper, we describe a comprehensive evaluation methodology which addresses both the underlying performance and the subjective usability of a tool. We present the key outcomes of a recently completed international evaluation campaign which adopted this approach and thus identify a number of new requirements for semantic search tools from both the perspective of the underlying technology as well as the user experience.

1 Introduction and Related Work

State-of-the-art semantic search approaches are characterised by their high level of diversity both in their features as well as their capabilities. Such approaches employ different styles for accepting the user query (e.g., forms, graphs, keywords) and apply a range of different strategies during processing and execution of the queries. They also differ in the format and content of the results presented to the user. All of these factors influence the user's perceived performance and usability of the tool. This highlights the need for a formalised and consistent evaluation which is capable of dealing with this diversity. It is essential that we do not forget that searching is a user-centric process and that the evaluation mechanism should capture the usability of a particular approach.

One of the very first evaluation efforts in the field was conducted by Kaufmann to compare four different query interfaces [1]. Three were based on natural language input (with one employing a restricted query formulation grammar);

* This work was supported by the European Union 7th FP ICT based e-Infrastructures Project SEALS (Semantic Evaluation at Large Scale, FP7-238975).

the fourth employed a formal query approach which was hidden from the end user by a graphical query interface. Recently, evaluating semantic search approaches gained more attention both in IR – within its most established evaluation conference TREC – [2] as well as in the Semantic Web community (SemSearch [3] and QALD[3] challenges).

The above evaluations are all based upon the Cranfield methodology [4][4]: using a test collection, a set of tasks and a set of relevance judgments. This leaves aside aspects of user-oriented evaluations concerned with the usability of the evaluated systems and the user experience which is as important as assessing the performance of the systems. Additionally, the above attempts are separate efforts lacking standardised evaluation approaches and measures. Indeed, Halpin et al. [3] note that "the lack of standardised evaluation has become a serious bottleneck to further progress in this field".

The first part of this paper describes an evaluation methodology for assessing and comparing the strengths and weaknesses of user-focussed Semantic Search approaches. We describe the dataset and questions used in the evaluation and discuss the results of the usability study. The analysis and feedback from this evaluation are described. The second part of the paper identifies a number of new requirements for search approaches based upon the outcomes of the evaluation and analysis of the current state-of-the-art.

2 Evaluation Design

In the Semantic Web community, semantic search is widely used to refer to a number of different categories of systems:

- *gateways* (e.g., Sindice [5] and Watson [6]) locating ontologies and documents
- approaches reasoning over data and information located within documents and ontologies (PowerAqua [7] and Freya [8])
- view-based interfaces allowing users to explore the search space while formulating their queries (Semantic Crystal [9], K-Search [10] and Smeagol [11])
- mashups integrating data from different sources to provide rich descriptions about Semantic Web objects (Sig.ma [12]).

The evaluation described here focuses on user-centric semantic search tools (e.g. query given as keywords or natural language or using a form or a graph) querying a repository of semantic data and returning answers extracted from them. The tools' results presentation is not limited to a specific style (e.g., list of entity URIs or a visualisation of the results). However, the results returned must be answers rather than documents matching the given query.

Search is a user-centric activity; therefore, it is important to emphasise the users' experience. An important aspect of this is the formal gathering of feedback from the participants which should be achieved using standard questionnaires. Furthermore, the use of an additional demographics questionnaire allows more

[3] http://www.sc.cit-ec.uni-bielefeld.de/qald-1

[4] http://www.sigir.org/museum/pdfs/ASLIB%20CRANFIELD%20RESEARCH%20PROJECT-1960/pdfs/

in-depth findings to be identified (e.g., if a particular type of user prefers a particular search approach).

2.1 Datasets and Questions

Subjects are asked to reformulate a set of questions using a tool's interface. Thus, it is important that the data set would be from an understandable and well-known domain (and hence, easily understandable by non-expert users) and, preferably, already have a set of questions and associated groundtruths. The geographical dataset from the Mooney Natural Language Learning Data[5] satisfies these requirements and has been used in a number of usability studies [1, 8]. Although the Mooney dataset is different from ones currently found on the Semantic Web such as DBpedia in terms of size, heterogeneity and quality, the assessment of the tools ability to handle these aspects is not the focus of this phase but rather the usability of the tools and the user experience.

The questions [13] used in the first evaluation campaign were generated based on the existing templates within the Mooney dataset. These contained questions with varying complexity and assessing different features. For instance, they contained simple with only 1 unknown concept such as "Give me all the capitals of the USA?" and comparative questions such as "Which rivers in Arkansas are longer than Aleghany river".

2.2 Criteria and Analyses

Usability Different input styles (e.g., form-based, NL, etc.) can be compared with respect to the input query language's expressiveness and usability. These concepts are assessed by capturing feedback regarding the user experience and the usefulness of the query language in supporting users to express their information needs and formulate searches [14]. Additionally, the expressive power of a query language specifies what queries a user is able to pose [15]. The usability is further assessed with respect to results presentation and suitability of the returned answers (data) to the casual users as perceived by them. The datasets and associated questions were designed to fully investigate these issues.

Performance Users are familiar with the performance of commercial search engines (e.g., Google) in which results are returned within fractions of a second; therefore, it is a core criterion to measure the tool's performance with respect to the speed of execution.

Analyses The experiment was controlled using custom-written software which allowed each experiment run to be orchestrated and timings and results to be captured. The results included the actual result set returned by a tool for a query, the time required to execute a query, the number of attempts required

[5] http://www.cs.utexas.edu/users/ml/nldata.html

by a user to obtain a satisfactory answer as well as the time required to formulate the query. We used post-search questionnaires to collect data regarding the user experience and satisfaction with the tool. Three different types of online questionnaires were used which serve different purposes. The first is the System Usability Scale (SUS) questionnaire [16]. The test consists of ten normalized questions and covers a variety of usability aspects, such as the need for support, training, and complexity and has proven to be very useful when investigating interface usability [17]. We developed a second, extended, questionnaire which includes further questions regarding the satisfaction of the users. This encompasses the design of the tool, the input query language, the tool's feedback, and the user's emotional state during the work with the tool. Finally, a demographics questionnaire collected information regarding the participants.

3 Evaluation Execution and Results

The evaluation consisted of tools from form-based, controlled-NL-based and free-NL-based approaches. Each tool was evaluated with 10 subjects (except K-Search [10] which had 8) totalling 38 subjects (26 males, 12 females) aged between 20 and 35 years old. They consisted of 28 students and 10 researchers drawn from the University population. Subjects rated their knowledge of the Semantic Web with 6 reporting their knowledge to be advanced, 5 good, 9 average, 10 little and 8 having no experience. In addition, their knowledge of query languages was recorded, with 5 stating their knowledge to be advanced, 12 good, 8 average, 6 little and 7 having no experience.

Firstly, the subjects were presented with a short introduction to the experiment itself such as why the experiment is taking place, what is being tested, how the experiment will be executed, etc. Then the tool itself was explained to the subjects; they learnt about the type and the functionality of the tool and how to apply it's specific query language to answer the given tasks. The users were then given sample tasks to test their understanding of the previous phases. After that, the subjects did the actual experiment: using the tool's interface to formulate each question and get the answers. Having finished all the questions, they were presented with the three questionnaires (Section 2.2). Finally, the subjects had the chance to talk about important and open questions and give more feedback and input to their satisfaction or problems with the system being tested.

Table 1 shows the results for the four tools participating in this phase. The *mean number of attempts* shows how many times the user had to reformulate their query in order to obtain answers with which they were satisfied (or indicated that they were confident a suitable answer could not be found). This latter distinction between finding the appropriate answer and the user 'giving up' after a number of attempts is shown by the *mean answer found rate*. *Input time* refers to the amount of time the subject spent formulating their query using the tool's interface, which acts as a core indicator of the tool's usability.

According to the ratings of SUS scores [18], none of the four participating tools fell in either the best or worst category. Only one of the tools (PowerAqua [7]) had a 'Good' rating with a SUS score of 72.25, other two tools

Table 1. Evaluation results showing the tools performance. Rows refer to particular metrics.

Criterion	K-Search	Ginseng	NLP-Reduce	PowerAqua
	Form-based	*Controlled NL-based*	*NL-based*	*NL-based*
Mean experiment time (s)	4313.84	3612.12	4798.58	2003.9
Mean SUS (%)	44.38	40	25.94	72.25
Mean ext. questionnaire (%)	47.29	45	44.63	80.67
Mean number of attempts	2.37	2.03	5.54	2.01
Mean answer found rate	0.41	0.19	0.21	0.55
Mean execution time (s)	0.44	0.51	0.51	11
Mean input time (s)	69.11	81.63	29	16.03

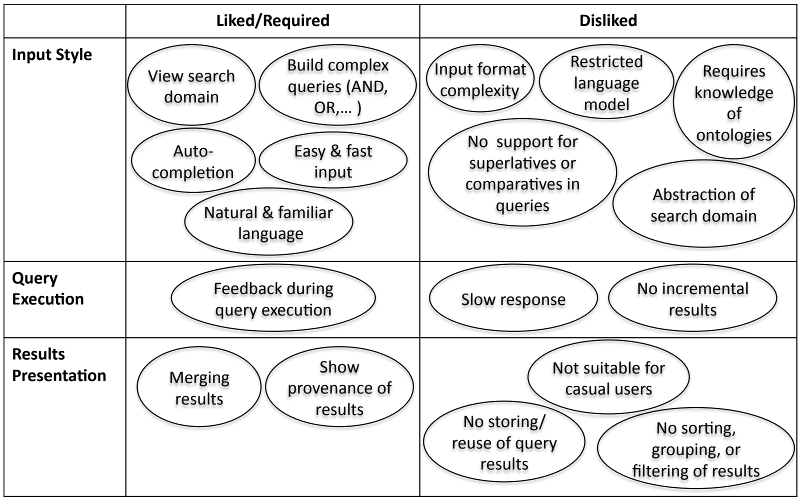
(Ginseng [19] and K-Search [10]) fell in the 'Poor' rating while the last one (Nlp-Reduce [20]) was classified as 'Awful'. The results of the questionnaires were confirmed by the recorded usability measures. Subjects using the tool with the lowest SUS score (Nlp-Reduce) required more than twice the number of attempts of the other tools before they were satisfied with the answer or moved on. Similarly, subjects using the two tools with the highest SUS and extended scores (PowerAqua and K-Search) found satisfying answers to their queries twice the times as for the other tools. Altogether, this confirms the reliability of the results and the feedback of the users and also the conclusions based on them.

4 Usability Feedback and Analysis

This section discusses the results and feedback collected from the subjects of the usability study. Figure 1 summarises the features most liked and disliked based on their feedback.

4.1 Input Style

On the one hand, Uren et al. [14] state that forms can be helpful to explore the search space when it is unknown to the users. Additionally, Corese [21] – which uses a form-based interface to allow users to build their queries – received very positive comments from its users among which was an appreciation for its form-based interface. On the other hand, Lei et al. [22] see this exploration as a burden on users that requires them to be (or become) familiar with the underlying ontology and semantic data. The results of our evaluation and the feedback from the users support both arguments. Additionally, we found that form-based interfaces allow users to build more complex queries than the natural language interfaces. However, building queries by exploring the search space is usually time consuming especially as the ontology gets larger or the query gets more complex. This was shown by Kaufmann et al. [1] in their usability study which found that users spent the most time when working with the graph-based system *Semantic Crystal*. Our evaluation supports this general conclusion: subjects using the form-based approach took between two to three times the time taken by users of natural language approaches. However, our analysis suggests a more nuanced behaviour. While freeform natural language interfaces are generally faster in terms of query formulation, we found this did not hold for approaches employing



	Liked/Required	Disliked
Input Style	View search domain Build complex queries (AND, OR,…) Auto-completion Easy & fast input Natural & familiar language	Input format complexity Restricted language model Requires knowledge of ontologies No support for superlatives or comparatives in queries Abstraction of search domain
Query Execution	Feedback during query execution	Slow response No incremental results
Results Presentation	Merging results Show provenance of results	Not suitable for casual users No storing/ reuse of query results No sorting, grouping, or filtering of results

Fig. 1. Summary of evaluation feedback: features most liked and disliked by users categorised with respect to query format, query execution, and results presentation.

a very restricted language model. For instance, query formulation took longer using Ginseng (restricted natural language) than K-Search (form-based). This is further supported by user feedback in which it was reported that they would prefer typing a natural language query because it is faster than forms or graphs.

Kaufmann et al. [1] also showed that a natural language interface was judged by users to be the most useful and best liked. Their conclusion, that this was because users can communicate their information needs far more effectively when using a familiar and natural input style, is supported by our findings. The same study found that people can express more semantics when they use full sentences as opposed to simply keywords. Similarly, Demidova et al. [23] state that natural language queries offer users more expressivity to describe their information needs than keywords – a finding also confirmed by the user feedback from our study.

However, natural language approaches suffer from both syntactic as well as semantic ambiguities. This makes the overall performance of such approaches heavily dependent upon the performance of the underlying natural language processing techniques responsible for parsing and analysing the users' natural language sentences. This was shown by the feedback we received from users of one of the natural language-based tools, one of which was "the response is very dependent on the use of the correct terms in the query". This was also confirmed by that approach achieving the lowest precision. Another limitation faced by the natural language approach is the lack of knowledge of the underlying ontology terms and relations by the users due to the high abstraction of the search domain. The effect of this is that any keywords or terms used by users are likely to be very different from the semantically-corresponding terms in the ontology. This in turn increases the difficulty of parsing the user query and affects the performance.

Using a restricted grammar as employed by Ginseng is an approach to limit the impact of both of these problems. The 'autocompletion' provided by the system based on the underlying grammar attempts to bridge the domain abstraction gap and also resembles the form-based approach in helping the user to better understand the search space. Although it provides the user with knowledge regarding which concepts, relations and instances are found in the search space and hence can be used to build valid queries, it still lacks the power of visualising the structure of the used ontology. The impact of this 'intermediate' functionality can be observed in the user feedback with a lower degree of dissatisfaction regarding the ability to conceptualise the underlying data but still not completely eliminated. The restricted language model also prevents unacceptable/invalid queries in the used grammar by employing a guided input natural language approach. However, only accepting specific concepts and relations – found in the grammar – limits the flexibility and expressiveness of the user queries. User coercion into following predefined sentence structures proves to be frustrating and too complicated [1, 24].

The feedback from the questionnaires showed that using superlatives or comparatives in the user queries (e.g.: highest point, longer than) was not supported by any of the participating tools; an issue raised by 8 subjects in the answer of the SUS question "What didn't you like about the system and why?" and by others in the open feedback after the experiment. Only one provided a feature similar to this functionality: the ability to specify a range of values for numeric datatypes. A comparative such as *less than 5000* could then be translated to the range *0 to 5000*. However, this was deemed to be both confusing (since the user had to decide what to use as the non-specified bound) and, when the non-specified bounds were incorrect, having a negative impact on the results.

4.2 Query Execution and Response Time

Speed of response is an important factor for users since they are used to the performance of commercial search engines (e.g., Google) in which results are returned within fractions of a second. Many users in our study were expecting similar performance from the semantic search tools. Although the average response time of three of the tools (K-Search, NLP-Reduce, Ginseng) is less than a second (44ms, 51ms, and 51ms respectively), users reported their dissatisfaction with these timings especially the ones who evaluated PowerAqua with response time of 11 seconds on average. The lack of feedback on the status of the execution process only served to increase the sense of dissatisfaction: no tool indicated the execution progress or whether a problem had occurred in the system. This lack of feedback resulted in users suspecting that something had gone wrong with the system – even if the search was still progressing– and start a new search. Furthermore, some tools made it impossible to distinguish between an empty result set, a problem with the query formulation or a problem with the search. This not only affected the users experience and satisfaction but also the approach's measured performance.

4.3 Results Presentation

Semantic Search tools are different from Semantic Web gateways or entry points such as Watson and Sindice. The latter are not intended for casual users but for other applications or the Semantic Web community to locate Semantic Web resources such as ontologies or Semantic Web documents and are usually presented as a set of URIs. For example, Sindice shows the URIs of documents and, for every document, it additionally presents the triples contained within the document, an RDF graph of the triples, and the used ontologies. Semantic Search tools are, on the other hand, used by casual users (i.e., users who may be experts in the domain of the underlying data but may have no knowledge of semantic technologies). Such users usually have different requirements and expectations of *what* and *how* results should be presented to them.

In contrast to these 'casual user' requirements, a number of the search tools did not present their results in a user-friendly manner and this was reflected in the feedback. Two approaches presented the full URIs together with the concepts in the ontology that were matched with the terms in the user query. Another used the instance labels to provide a natural language presentation; however, such labels (e.g., 'montgomeryAl') were not necessarily suitable for direct inclusion into a natural language phrase. Indeed, the tool also displayed the ontologies used as well as the mappings that were found between the ontology and the query terms. Although potentially useful to an expert in the semantic web field, this was not helpful to casual users.

The other commonly reported limitation of all the tools was the degree to which a query's results could be stored or reused. A number of the questions used in the evaluation had a high complexity level and needed to be split into two or more sub-queries. For instance, for the question "Which rivers in Arkansas are longer than the Allegheny river?", the users were first querying the data for the length of the Allegheny river and then performing a second query to find the rivers in Arkansas which are longer than the answer they got. Therefore, users often wanted to use previous results as the basis of a further query or to temporarily store the results in order to perform an intersection or union operation with the current result set. Unfortunately, this was not supported by any of the participating tools. However, this shows that users have very high expectations of the usability and functionalities offered by a semantic search tool as this requirement is not provided even by traditional search systems (e.g., Google and Yahoo). Another means of managing the results that users requested was the ability to filter results according to some suitable criteria and checking the provenance of the results; only one tool provided the latter. Indeed, even basic manipulations such as sorting were requested – a feature of particular importance for tools which did not allow query formulations to include superlatives.

5 Future Directions

This section identifies a number of areas for improvement for semantic search tools from the perspective of the underlying technology and the user experience.

5.1 Input Style

Usability The feedback shows that it's very helpful for users – especially those who are unfamiliar with the underlying data – to explore the search space while building their queries using view-based interfaces which expose the structure of the ontology in a graphical manner. It gives users a much better understanding of what information is available and what queries are supported by the tool. In contrast, the feedback also shows that, when creating their queries, users prefer natural language interfaces because they are quick and easy. Clearly both approaches have their advantages; however, they suffer from various limitations when used separately as discussed in Sec. 4.1. Therefore, we believe that the combination of both approaches would help get the best of both worlds.

Users not familiar with the search domain can use a form-based or natural language-based interface to build their queries. Simultaneously, the tool should dynamically generate a visual representation of the user's query based upon the structure of the ontology. Indeed, the user should be able to move from one query formulation style to another – at will – with each being updated to reflect changes made in the other. This 'dual' query formulation would ensure a casual user correctly formulates their intended query. Expert users, or those who find it laborious to use the visual approach, would simply use the natural language input facility provided by the tool. An additional feature for natural language input would be an *optional* 'auto-completion' feature which could guide the user to query completion given knowledge of the underlying ontology.

Expressiveness The feedback also shows that the evaluated tools had difficulties with supporting complex queries such as the ones containing logical operators (e.g, "AND"). Allowing the user to input more than one query and combining them with their chosen logical operator from a list included in the interface would reduce the impact of this limitation. The tool would merge the results according to the used operator (e.g., "intersection" for "AND"). For instance, a query such as "What are the rivers that pass through California and Arizona?" would be constructed as two subqueries: "What are the rivers that pass through California?" and "What are the rivers that pass through Arizona?" with the final results being the intersection of both result sets.

Furthermore, the evaluated tools faced similar difficulties with supporting superlatives and comparatives in users' queries. Freya [8] deals with this problem by asking the user to identify the correct choice from a list of suggestions. To illustrate this we'll use the query *"Which city has the largest population in California?"*. If the system captures a concept in the user query that is a datatype property of type number, it generates maximum, minimum and sum functions. The user can then choose the correct superlative or comparative depending on their needs. A similar approach can be used to allow the use of superlatives and comparatives in natural language interfaces and form-based interface. In the case of the latter, whenever a datatype property is selected by the user, the tool can allow them to select from a list of functions that cover superlatives and comparatives (e.g., 'maximum', 'minimum', 'more than', 'less than').

5.2 Query Execution and Response Time

Several users reported dissatisfaction with the tools' response time to some of their queries. Users appreciated the fact that the tools returned more accurate answers than they would get from traditional search engines, however this did not remove the effect of the delay in response – even if it was relatively small. Additionally, the study found that the use of feedback reduces the effect of the delay; users showed greater willingness to wait if they were informed that the search is still being performed and that the delay is not due to a failure in the system. The presentation of intermediate, or partially complete, results reduces the perceived delay associated with the complete result set (e.g., Sig.ma [12]). Although only partial results are available initially, it provides both feedback that the search is executing properly and allows the user to start thinking about the content of the results before the complete set is ready. However, it ought to be noted that this approach may induce confusion in the user as the screen content changes rapidly for a number of seconds. Adequate feedback is essential even for tools which exhibit high performance and good response times. Delays may occur at a number of points in the search process and may be the result of influences beyond the developer's control (e.g., network communication delays).

5.3 Results Presentation

Most of the users were frustrated by the fact that they didn't understand the results presented to them, feeling that too much technical knowledge was assumed. The evaluation shows that the tools underestimated the effect of this on the users' experience and satisfaction.

Query answers ought to be presented to users in an accessible and attractive manner. Indeed, the tool should go a step further and augment the direct answer with associated information in order to provide a 'richer' experience for the user. This approach is adopted by WolframAlpha[6]; for example, in response to '*What is the capital of Alabama?*' WolframAlpha includes the natural language presentation of the answer as well as various population statistics, a map showing the location of the city, and other related information such as the current local time, weather and nearby cities.

An interesting requirement found by our study was the ability to store the result set of a query to use in subsequent queries. This would allow more complex questions to be answered which, in turn, improves the tools' performance. QuiKey [25] provides a functionality similar to this. QuiKey is an interaction approach that offers interactive fine grained access to structured information sources in a lightweight user interface. It allows a query to be saved which can later be used for building other queries. More complex queries can be constructed by combining saved queries with logical operators such as 'AND' and 'OR'.

Result management was also identified as being of importance to users with commonly requested functionality included sorting, filtering and more complex activities such as establishing the provenance and trustworthiness of certain results. For example, Sig.ma [12] creates information aggregates called Entity Profiles and provides users with various capabilities to organise, use and establish the provenance of the results. Users can see all the sources contributing to a specific profile and approve or reject certain ones thus filtering the results. They can

[6] http://www.wolframalpha.com/

also check which values in the profile are given by a specific source thus checking provenance of the results. Sig.ma also supports the aspect of merging separate results by allowing users to view ones returned only from selected sources.

6 Conclusions

We have presented a flexible and comprehensive methodology for evaluating different semantic search approaches; we have also highlighted a number of empirical findings from an international semantic search evaluation campaign based upon this methodology. Finally, based upon analysis of the evaluation outcomes, we have described a number of additional requirements for current and future semantic search solutions.

In contrast to other benchmarking efforts, we emphasised the need for an evaluation methodology which addressed both performance and usability [24]. We presented the core criteria that must be evaluated together with a discussion of the main outcomes. This analysis identified two core findings which impact upon semantic search tool requirements.

Firstly, we found that an intelligent combination of natural language and view-based input styles would provide a significant increase in search effectiveness and user satisfaction. Such a 'dual' query formulation approach would combine the ease with which a view-based approach can be used to explore and learn the structure of the underlying data whilst still being able to exploit the efficiency and simplicity of a natural language interface.

Secondly, (and perhaps of greatest interest to users) was the need for more sophisticated results presentation and management. Results should allow a large degree of customisability (sorting, filtering, saving of intermediate results, augmenting, etc). Indeed, it would also be beneficial to provide data which is supplementary to the original query to increase 'richness'. Furthermore, users expect to be able to have immediate access to provenance information.

In summary, this paper has presented a number of important findings which are of interest both to semantic search tool developers but also designers of interactive search evaluations. Such evaluations (and the associated analyses as presented here) provide the impetus to improve search solutions and enhance the user experience.

References

1. Kaufmann, E.: Talking to the Semantic Web — Natural Language Query Interfaces for Casual End-Users. PhD thesis, University of Zurich (2007)
2. Balog, K., Serdyukov, P., de Vries, A.P.: Overview of the TREC 2011 Entity Track. In: TREC 2011 Working Notes
3. Halpin, H., Herzig, D.M., Mika, P., Blanco, R., Pound, J., Thompson, H.S., Tran, D.T.: Evaluating Ad-Hoc Object Retrieval. In: Proc. IWEST 2010 Workshop
4. Cleverdon, C.W.: Report on the first stage of an investigation onto the comparative efficiency of indexing systems. Technical report, The College of Aeronautics, Cranfield, England (1960)

5. Tummarello, G., Oren, E., Delbru, R.: Sindice.com: Weaving the Open Linked Data. In: Proc. ISWC/ASWC 2007
6. d'Aquin, M., Baldassarre, C., Gridinoc, L., Angeletou, S., Sabou, M., Motta, E.: Characterizing Knowledge on the Semantic Web with Watson. In: EON. (2007) 1–10
7. Lopez, V., Motta, E., Uren, V.: PowerAqua: Fishing the Semantic Web. In: The Semantic Web: Research and Applications. (2006) 393–410
8. Damljanovic, D., Agatonovic, M., Cunningham, H.: Natural Language Interface to Ontologies: combining syntactic analysis and ontology-based lookup through the user interaction. In: Proc. ESWC. (2010)
9. Bernstein, A., Kaufmann, E., Göhring, A., Kiefer, C.: Querying Ontologies: A Controlled English Interface for End-users. In: Proc. ISWC 2005
10. Bhagdev, R., Chapman, S., Ciravegna, F., Lanfranchi, V., Petrelli, D. In: Proc.. ESWC 2008
11. Clemmer, A., Davies, S.: Smeagol: A specific-to-general semantic web query interface paradigm for novices. In: Proc. DEXA 2011
12. Tummarello, G., Cyganiak, R., Catasta, M., Danielczyk, S., Delbru, R., Decker, S.: Sig.ma: live views on the web of data. In: Proc. WWW 2010
13. Wrigley, S.N., Elbedweihy, K., Reinhard, D., Bernstein, A., Ciravegna, F.: D13.3 Results of the first evaluation of semantic search tools. Technical report, SEALS Consortium (2010)
14. Uren, V., Lei, Y., Lopez, V., Liu, H., Motta, E., Giordanino, M.: The usability of semantic search tools: a review. The Knowledge Eng. Rev. **22** (2007) 361–377
15. Angles, R., Gutierrez, C.: The Expressive Power of SPARQL. In: Proc. ISWC 2008
16. Brooke, J.: SUS: a quick and dirty usability scale. In: Usability Evaluation in Industry. (1996) 189–194
17. Bangor, A., Kortum, P.T., Miller, J.T.: An Empirical Evaluation of the System Usability Scale. Int't J. Human-Computer Interaction **24**(6) (2008) 574–594
18. Bangor, A., Kortum, P.T., Miller, J.T.: Determining what individual SUS scores mean: Adding an adjective rating scale. J. Usability Studies **4**(3) (2009) 114–123
19. Bernstein, A., Kaufmann, E., Kaiser, C.: Querying the Semantic Web with Ginseng: A Guided Input Natural Language Search Engine. In: Proc. WITS 2005 Workshop
20. Kaufmann, E., Bernstein, A., Fischer, L.: NLP-Reduce: A "naïve" but Domain-independent Natural Language Interface for Querying Ontologies. In: Proc. ESWC 2007
21. Corby, O., Dieng-Kuntz, R., Faron-Zucker, C., Gandon, F.: Searching the Semantic Web: Approximate Query Processing Based on Ontologies. IEEE Intelligent Systems **21** (2006) 20–27
22. Lei, Y., Uren, V., Motta, E.: SemSearch: A Search Engine for the Semantic Web. In: Proc. EKAW2006
23. Demidova, E., Nejdl, W.: Usability and Expressiveness in Database Keyword Search : Bridging the Gap. In: Proc. VLDB PhD Workshop. (2009)
24. Wrigley, S.N., Elbedweihy, K., Reinhard, D., Bernstein, A., Ciravegna, F.: Evaluating Semantic Search Tools using the SEALS platform. In: Proc. IWEST 2010 Workshop
25. Haller, H.: QuiKey - An Efficient Semantic Command Line. In: Proc. EKAW 2010