Black Box Evaluation for Operational Information Retrieval Applications

Martin Braschler, Melanie Imhof, and Stefan Rietberger

Zurich University of Applied Sciences, Winterthur, Switzerland
{bram,imhf,riet}@zhaw.ch

Abstract. The black box application evaluation methodology described in this tutorial is applicable to a broad range of operational information retrieval (IR) applications. Contrary to popular, traditional IR evaluation approaches that are limited to measure the IR system performance on a test collection, the black box evaluation methodology considers an IR application in its entirety: the underlying system, the corresponding document collection, and its configuration/application layer. A comprehensive set of quality criteria is used to estimate the user's perception of the application. Scores are assigned as a weighted average of results from tests that evaluate individual aspects. The methodology was validated in a small evaluation campaign. An analysis of this campaign shows a correlation between the testers' perception of the applications and the evaluation scores. Moreover, functional weaknesses of the tested IR applications can be identified and then systematically targeted.

Keywords: information retrieval, application evaluation, black box, user perception.

1 Introduction

This tutorial paper explores a method to evaluate the quality of operational information retrieval (IR) applications. For the purpose of this paper, we define an IR application to consist of an IR system, a specific document collection (document base), a business application layer (including front-end), and a configuration set.

Traditionally, IR evaluation has concentrated on measuring the retrieval effectiveness of IR systems. The ranked list retrieved by an IR system is compared to the relevance of each document in a fixed test collection with respect to a query. However, such measurement ignores several important aspects of entire IR applications as defined above, which we expect to (sometimes strongly) affect the user's perception of (and, thus, satisfaction with) the application. For example, the user will not value a high retrieval effectiveness if the responsiveness of the IR system is too low.

The methodology presented herein employs a black box approach. It aims at practitioners, who conduct the evaluation "in the wild"; i.e. on an operational system. We have further explored how to adapt the methodology to different application domains, such as cultural heritage, search for innovation and medical image retrieval.

N. Ferro (Ed.): PROMISE Winter School 2013, LNCS 8173, pp. 192–207, 2014.

Substantial parts of the methodology have been developed as part of the activities of the PROMISE EU FP7 Network of Excellence [1].

By employing this "black box application evaluation", we perform comparative evaluation based on an estimate of user perception. The choice of the notion of "user perception" fits the limitation of only being able to assess aspects of the IR application which typical users can access and experience (due to the black box nature of the approach). More importantly, however, we also feel that the targeted audience of such evaluation results (e.g. corporate decision makers) has an interest in assessing and improving the user perception of these applications.

2 Related Work

The evaluation of IR systems in the narrower sense (systems for ranked retrieval) is a well-researched field, and mature methods are widely employed. To briefly summarize the most important approach, it helps to reflect the basic problem addressed by IR systems. The goal of IR systems is "[…] to retrieve all and only those documents that the inquiring patron wants (or would want)" [2]. This is a difficult problem for a number of reasons. Typically, IR systems allow access to large, potentially heterogeneous, document collections that contain unstructured free-text (or, in the case of multimedia IR systems, non-textual items). The documents are usually written by a range of authors, and can stem from a variety of sources. These authors have considerable freedom in expressing information: there is no set vocabulary, and paraphrasing, metaphors etc. are used. Linguistic phenomena such as homonyms, synonyms, morphology etc. complicate matters further. Users search such a body of documents based on "information needs" - aiming to solve problems for which they are missing information. It would be paradox to expect the users to be able to form perfect queries: they would have to effectively "predict" the formulation used by the author of a matching document. This would require the user to read the document itself – before it was found. As a consequence, an "exact match" strategy (as used in database management systems), whereby the system matches a set of keywords exactly as entered with the documents, is rarely an effective strategy for information retrieval. "Best match" strategies dominate IR approaches, where query terms are weighted, and retrieval scores RSV(q,dj) (the retrieval status value for document j given query q) are calculated. Instead of returning a set of (exactly) matching documents, a ranked list sorted in descending order of the RSV scores is returned. The effectiveness of obtaining these "best matches" (the "retrieval effectiveness") directly depends on the mechanisms employed for processing documents and queries, and for later matching them. These mechanisms have to consider all the phenomena described above. However, further complicating things is a subjective notion of how users would judge the relevance of these partially matching items that are returned by the system. The same document may well be judged as either relevant or irrelevant by different users with respect to the same query, depending on the context, background, or personal preferences.

No retrieval mechanisms can therefore in practice deliver optimal results (i.e. all relevant items, and only relevant items, for all queries). The different popular strategies, such as vector-space retrieval, probabilistic retrieval, language models, etc. present the user with differing results that need to be assessed for effectiveness. It is thus not surprising that evaluation of information retrieval (IR) systems is an extensively researched problem. Starting all the way back in the 1960s the foundation for today's most prominent IR system evaluation methodology ("the Cranfield paradigm") was laid [3]. In a nutshell, the evaluation is conducted in a "lab style" environment, where the documents are fixed (in the form of a "test collection"), and the users are abstracted (in the form of "information needs", which are attributed to those users). To conduct a retrieval experiment using the Cranfield paradigm, queries are derived from the information needs, and then run against an index of the documents. The scalability of this approach is limited directly by the capacity to judge the results – conceptually, every document in the collection has to be assessed for relevance for every query – i.e. an effort of the order number of queries times number of documents[1].

The Cranfield paradigm has been highly successful in driving progress in the academic field of information retrieval, substantially aiding the development of more effective term weighting schemes, stemmer components, indexing pipelines, among others. Catalyst for this has been the formation of large evaluation campaigns that bundle the evaluation efforts of IR academics and system developers worldwide (TREC, CLEF, NTCIR, FIRE, etc.). Despite its success, the paradigm's applicability for evaluation of entire IR applications is limited, since retrieval effectiveness is but one aspect that influences a user's perception of IR applications.

Log file analysis is an alternative strategy to evaluate search engines. Transaction logs collect significant amounts of user behavior such as their clicks and queries. Later the logs are used to evaluate the quality of the search engine [5]. However the users' perception is only deduced from the logs [6].

Two IR applications can be compared using A/B testing, where users are randomly assigned to one of two systems [7]. By analyzing user behavior it can be seen which system is preferred. A very similar evaluation methodology was suggested by Radlinski. Instead of assigning users to one of two search engines, the result lists of both engines are interleaved and presented [8].

User based evaluation aims to measure user perception. Dunlop's evaluation framework accounts for user experience by evaluating surface interactions and system usability [9]. Borlund's work on interactive information retrieval (IIR) describes how to measure IR application performance when considering the humans cognitive perspective [10].

The methodology for "black box application evaluation" discussed in this tutorial paper is partially based on earlier work that is presented in two studies [11; 12]. Those studies describe an evaluation based on a grid of scripted tests in an attempt to identify the state of Swiss and German enterprise Web portal search, and are much

[1] There are encouraging signs that in spite of the subjectivity of relevance, multiple assessments of relevance per document/query pair are not necessary for many evaluation scenarios.[4].

narrower in focus We adapted the main criteria categories from this earlier work, whereas the individual tests themselves were developed from scratch. The previous studies furthermore omit the discussion of different domains and use case scenarios and do not explore the question of the underlying measures in detail, both discussion topics of the present report.

3 Black Box Application Evaluation

The methodology aims to evaluate entire IR applications without any further knowledge of their inner workings or the components employed. This makes the methodology broadly applicable to a large range of IR applications. Further, it eases the use of the methodology on live, operational applications, which was an important design goal ("evaluation in the wild"). Specifically, the main guidelines for the evaluation are:

1. The evaluation is performed in a "black box mode", or minimally invasive
2. The evaluation is performed on operational applications ("in the wild")
3. The evaluation is performed in a clearly defined use case domain context

The third guideline determines the applicability of different tests employed during evaluation. Varying influence of different criteria on user perception may make comparison across different use case scenarios difficult. Information retrieval applications for the purpose of this paper are defined to consist of

1. a specific data/document collection
2. an information retrieval (IR) system, and
3. a business application/GUI layer,

as well as the specific configuration of these components. The following figure (figure 1) shows the named components and configuration, highlighting the latter's equal importance to operational performance.

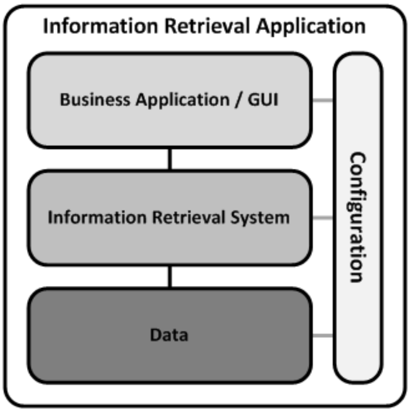
Information Retrieval Application
Business Application / GUI
Information Retrieval System
Data
Configuration

Fig. 1. Information Retrieval Application Model, based on [14]

For the purpose of the black box evaluation, we do not incorporate specific users directly into this model of information retrieval applications. Instead, prototypical users are modeled to the extent as their actions and their preferences are reflected in the criteria that are chosen to be evaluated, influencing the weight that each criterion has on the overall scores.

The modus operandi of the evaluation process is to employ a comprehensive set of all identified "quality" criteria that are believed to be tied to user perception or the user's search experience (at presence, 43 criteria). This is an ambitious goal: it is a large undertaking to identify and define the individual tests, elaborate the corresponding testing steps, and assign both scoring procedures and overall weights for later aggregation. Clearly, there is much room for the methodology to develop over time, as new insights are gained into many of the issues addressed by the tests. For the time being, we employ "simple" tests, which we organized hierarchically. For the present iteration of the methodology, the design goal was a maximum number of coarse, orthogonal tests that should ideally cover most aspects of the IR application that may influence the defined evaluation metric. Note that depending on the use case domains served by an IR application, the resulting hierarchical tree of tests may have to be pruned before evaluation, as a number of tests may not be applicable. The following figure (figure 2) shows a schematic view of the criteria/test hierarchy and applications when set up for an actual evaluation.

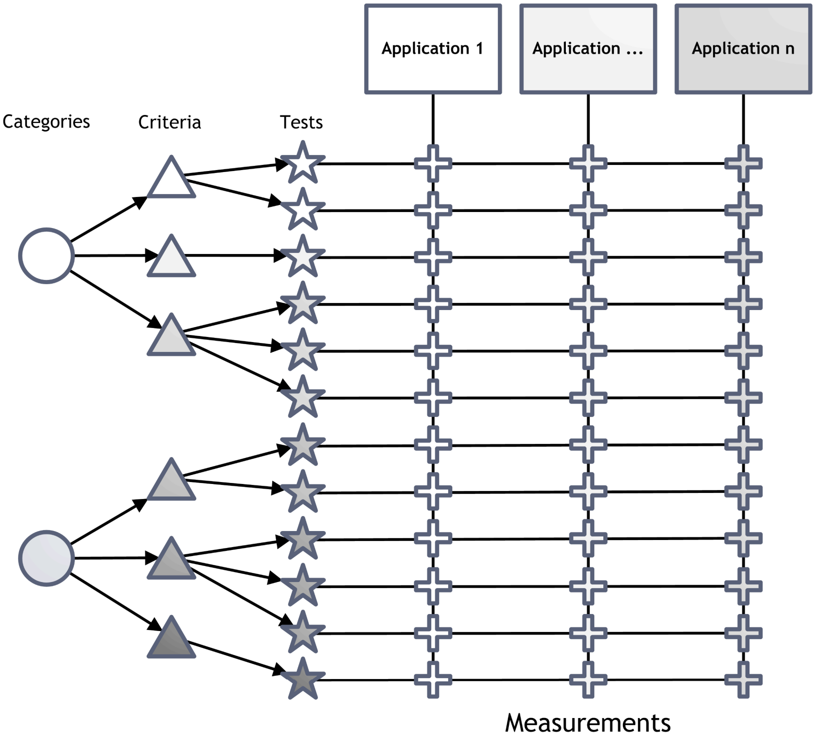

Fig. 2. Evaluation Grid for Multiple Applications

The methodology in its current form groups the criteria and associated tests into four main categories (derived from [12]):

1. **Indexing**: Contains all tests addressing the indexing component of an IR application, specifically how documents are processed and stored to allow later retrieval
2. **Matching**: Contains all tests covering the matching between queries and documents
3. **User Interface**: Contains all tests that address user interface criteria, such as presentation, usability and others
4. **Search Result**: Contains all tests that address the "quality" of search results, such as overall retrieval effectiveness

4 Conducting the Evaluation

The evaluation is conducted according to a test script, containing all necessary information for performing the individual tests. The script contains step-by-step instructions for the tester that are designed to minimize the necessity of testers to resort to "creative" testing, i.e. the outcome of the tests should ideally be as independent as possible from the person of the actual tester. It is a stated goal to automate as many of the tests as possible in the future; however, for the time being, most tests contain steps that require intellectual effort, such as for example the selection of small excerpts from a document. The tests further contain clear definitions as to how to determine if a test can be conducted at all (abort conditions), and if so, how to score its outcomes. The complete reproduction of the test script lies outside of the scope of this paper. However, a comprehensive description of all tests can be found in [1].

To start with the evaluation, the set of tests is determined by pruning the individual tests that do not apply to the use case domain underlying the IR application. There is a field in each test description that gives indications on the applicability of the test to different domains. If the application addresses use case domains not yet considered in the development of the methodology, this step requires more effort. Discussions with domain experts need then be held, to assess the merits of the tests and their likely contributions to the overall user perception. Alternatively, some tests may be adapted to the new use case domain. Next, each test is carried out according to the script. It is important to carefully check the abort conditions given for each test: there are preconditions recorded for the tests which must be met to calculate corresponding scores.

Overall evaluation scores for IR applications are computed based on the aggregated scores of the individual criteria. The weights should be defined in advance based on the practical significance of any criterion in the evaluated application's use case domain. How to weight the individual criteria is still matter of on-going research. For the time being, we resort to uniform weighting across all criteria. Where tests operate on a different level of granularity, multiple, associated tests are bundled, and assigned a weight as a whole. Experience gathered so far seems to indicate that this "coarse" approach works well enough, possibly due to use of a large number of tests (see below for a discussion of the preliminary results of applying the methodology).

The calculated scores lend themselves directly to comparative evaluation of IR applications (or different incarnations of one particular IR application), but by choosing scoring methods that are based on absolute counts, there is a well-defined maximum score, which allows assessment and monitoring of single applications as well. Finally, it is possible to use the methodology in an evaluation campaign style, spanning many different IR applications, restricted by the use case domains that they serve.

5 Individual Tests

This section outlines the test description structure, a list of tests with short summaries and finally, selected examples of criteria and tests for each main category. The full description of all tests can be found in [1]. The individual tests are structured according to a template containing the following main sections (table 1):

Table 1. Test Description Structure

Section	Content
Assumption	Assumptions/preconditions for the tests to be valid. Also, in this field, the expected behavior (attributed to the preferences of the prototypical user) is described
Irregularity	A description of unwanted behavior tied to the test.
Root causes	A description of possible causes for irregularities
Test	Description of the actual step-by-step testing procedure, includes scoring and abort conditions
Use case domain adaption	Any information necessary to decide on adaptions for specific use cases and/or decide on the applicability of a test to a specific use case.

5.1 Criteria List

The following table (table 2) gives a full list of all criteria that have tests currently defined for the black box IR application evaluation methodology. The tests have been compiled in two steps. In a first step four main categories (indexing, matching, user interface and search results) have been adopted, analogous to earlier work [11; 12]. In a second step a board of use case domain stakeholders assembled the quality criteria for their domains. We are confident that this process has given us a reasonable base set of criteria. Most criteria have been included in the test script, with few exceptions, most notably a criterion for the evaluation of informational queries, where no simple mechanism for measurement has been found. We plan to publish the criteria list as a living document where stakeholders from research and industry can participate in order to have a broader basis in the future.

Table 2. Criteria List

Criterion Name	Description
INDEXING	
Completeness	Are all (browsable) documents findable through the search functionality?
Freshness	Are the newest documents findable through the search functionality?
Special Characters	Does the application handle diacritics and special characters correctly?
Tokenization	Are terms and names with complex punctuation and/or hyphenation treated correctly?
Decompounding	Are complex terms (such as used in agglutinative languages) handled correctly?
Named Entities	Are named entities handled and disambiguated correctly?
Stemming	Does the system normalize word forms?
Meta-Data Quality	Are meta data fields correct and complete?
Office Document Handling	Are binary office documents (PDF, Office formats etc.) handled correctly?
Separation of Actual Content and Representation	Are structural elements (such as headers, footers) excluded from searches for document content?
Duplicate (Content) Documents	Are duplicate entries removed from result lists?
MATCHING	
Query Syntax	Does the application offer query operators (e.g. Boolean operators)?
Phrasal Queries	Does the application offer phrasal querying (e.g. by using quotes)?
Over- and under-specified Queries	Are over- and under-specified queries (e.g. too many specific search terms or too few, too broad search terms) handled gracefully?
Feedback	Does the application allow the user to give feedback on a document's relevance, with the search result influenced by such feedback?
Multimedia	Does the application offer search for videos, images or audio content?
Cross-Language Information Retrieval	Does the application allow querying across different languages?
USER INTERFACE	
Performance/Responsiveness	Does the application provide fast response times?
Browsing	Are users able to efficiently navigate (browse) the content without using ad-hoc querying?

Table 2. (*continued*)

Field Search (Facets)	Can search results be filtered by categories?
Query Term Highlighting	Are matching query terms highlighted in documents?
Document Summarization	Are suitable document summarizations („snippets“? presented in the result lists?
Result List Presentation	Is the result list presentation well organized?
Exception Handling	Is the application stable?
Term Suggestions	Does the application provide term suggestions? (potentially for technical terms)
User Guidance	Are users assisted in query formulation?
Related Content	Is content related to searches automatically shown?
Context Information	Is context additional to the search results presented (e.g. derived from corpus statistics etc.)
Personalization	Does the application manage user profiles?
Localization	Is the application adapted to different regional audiences?
Result List Import/Export	Can search results be imported and/or exported?
Sorting of Result List	Can result lists be (re-)sorted according to metadata or other criteria?
Justification of Results	Is there any supplementary information on how results were generated? (explanation of weighting, of matches etc.)
Monitoring	Can long-standing queries be monitored over time?
System Override/User Control	Can features, such as spelling correction, stemming etc. be turned off?
Navigational Aids	Can users navigate between different queries?
Social Aspects	Can search results be shared with other users?
Entertainment/Fun	Is the user experience good?
Mobile Access	Is there a mobile version of the user interface?
SEARCH RESULT	
Navigational Queries	Can users easily locate “entry points” into subsections of the website?
Factual Queries	Can users effectively find factual information?
Known/Suspected Item Retrieval	Can users effectively (re-)find a document in the application that they have accessed before or expect to be present?
Diversity	Are different aspects of ambiguous queries covered in the search results?

It is beyond the scope of this paper to present the full version of the test script, detailing all the tests for the different criteria above. For the complete details, see [1]. We will restrict the discussion to four specific, illustrative examples (one per category) in the following.

We begin with a test designed to evaluate whether the IR application uses domain knowledge to treat named entities (often core business entities) in a special way. This test is filed in category “Indexing”. To execute the test, it is required that the tester has gained some knowledge about the underlying application domain in order to identify the named entities. The test is repeated for five named entities to compensate for named entities that are not handled. Resulting scores are in a range of 0 to 5.

Indexing Criterion Example: Named Entities

Assumption	Users want to search for named entities where the respective entity is very clearly defined within the context of the application. Inability to find documents pertaining to the entities at a high rank in the result list is disruptive to the user experience.
Irregularity	Clearly defined entities from the application context cannot be directly found using their names as a short query.
Root Cause	The document indexing process does not consider named entities and thus tokenizes them in less informative bits.
Test	1. Identify 5 named entities (preferably composed of 2 or more terms) based on the applications context. Usually you are able to deduce these from the content. (a) Abort if less than 5 named entities can be found 2. Search for the entities using only their name 3. Score success (0, 1, 2, 3, 4, 5) for each query which returns results that clearly refers to the correct entity, and not to other entities that share parts of the name.
Use Case Domain Adaptations	N/A

The second test we discuss focuses on the issue of over- and underspecified queries. It is filed in category “Matching”. The test can be carried out without any knowledge about the application domain or even information retrieval mechanisms. The score consists of two parts. One point each is given for correct handling of overspecified and underspecified queries, respectively. The resulting score for this test is in a range of 0 to 2.

Matching Criterion Example: Over- and Underspecified Queries	
Assumption	Users feel irritated if long queries return very few or no results and short queries return almost the entire collection.
Irregularity	Missing the application's unknown "sweet spot" in terms of query length returns an undesirable number of results. Users receive no indication of what went wrong**.**
Root Causes	• No user guidance when result set has an unusual number of hits • Matching model punishes verbose descriptions
Test	1. Copy and paste a sentence from any document within the application into a query and add some out-of-context terms 2. Score success (1) if the document can still be found, score failure (0) otherwise 3. Use 2 terms from the application's context as a query, which should return a very large number of results 4. Score success (1) if the application offers suggestions or facilities to improve your search, e.g. further terms, browsing, etc. Score failure (0) otherwise for a total of (0, 1, 2)
Use Case Domain Adaptations	N/A

The criterion on query term highlighting is an example for a set of criteria that test for the presence or absence of features. It is filed under category “User Interface”. The test script is easy to follow for a human, but hard to automate since the terms can be highlighted in different ways; e.g. color, bold, italic. The resulting score is either 0 or 1.

User Interface Example: Query Term Highlighting	
Assumption	Highlighted query terms in a result list help users to preliminarily assess the relevance of documents.
Irregularity	Query terms are not highlighted or otherwise marked in the result list.
Root Cause	Feature not implemented
Test	Score success (1) if query terms are marked in any way in the result list. Otherwise score failure (0).
Use Case Domain Adaptations	N/A

Lastly, we discuss a criterion filed under category “Search Result”. The criterion on factual queries is not applicable to the search for innovation use case. In that domain querying general facts leads to a lot of results, while querying very specific facts returns the document itself. However the search for a specific document is already covered in the known item retrieval criterion. Resulting scores are in a range of 0 to 5.

Search Result Example: Factual Queries	
Assumption	Users enter queries to find a single fact. A single trustworthy document is sufficient to satisfy the information need.
Irregularity	Factual information cannot be found by suitable queries.
Root Causes	• Freshness and completeness of index are lacking • Bad treatment of binary documents (e.g. PDF) • Missing document summaries or snippets in result list
Test	1. Pick 5 facts from the application's content, examples: (a) Company's year of incorporation (b) Number of branches (c) Revenue (d) CEO (e) Product lines (f) etc. 2. Build short queries for these facts from the context 3. Score success for each query which retrieved the sought for fact in the top 10 results (0, 1, 2, 3, 4, 5)
Use Case Domain Adaptations	Search for Innovation: Criterion not applicable. Cultural Heritage: Criterion not applicable

6 Validation of Methodology

To validate the methodology, a campaign was conducted by the Promise EU FP7 Network of Excellence to evaluate a number of public websites that offer search functionality, and thus qualify for our definition of an IR application. The websites were chosen according to the following criteria:

1. Only publicly accessible web sites were considered
2. The website offers search functionality, and functions as an IR application in the sense of the definition of this paper

3. The website fits one of the following four use case domains: “enterprise/extranet search”, “cultural heritage”, “search for innovation”, “visual clinical support”
4. The website addresses users in one of the countries represented by the PROMISE partners that conducted the tests: Germany, Switzerland, France, Italy and Sweden. Some exceptions were made for websites run by European organizations (mainly in order to get good coverage for the use case domains mentioned above)

In total, 62 websites conformant to the above criteria were evaluated. The time to evaluate a single website was roughly of the order of half a person day, i.e. approximately 4 working hours. In addition to following the test script and recording the respective scores, testers were also asked to record their user experience. They provided a coarse score (0 to 2) for the “fun” they had using the application and a more finely grained score (0 to 10) for their overall user experience. This gives the possibility to correlate this subjective experience by testers with the estimates of user perception calculated through the evaluation methodology. As a working title, the campaign was run jokingly under the title of “guerilla campaign”, to express the fact that any website can potentially be a target of this evaluation, with direct involvement by the operators not being necessary.

7 Results and Lessons from Guerrilla Campaign

Aside from validating the feasibility of the evaluation methodology, and giving input for improvements to the test script, the guerrilla campaign also gives insight into the state-of-the-art of public websites in the use case domains covered. Please note that this was not a primary motivation for the campaign, and we did not strive for the necessary “completeness” in the websites covered to get a real “overview of the state of the art”. The amount of websites evaluated was strictly limited by the effort that partners had available for validating the methodology itself. Even so, the number of websites is large enough to give interesting insights, and possibly guidance for later applications of the methodology by practitioners.

We present the overall results in the form of a boxplot in figure 3. The aggregated scores are given for the four main categories. For each category, it is therefore possible to read the maximum, minimum and median performance from the graph. To summarize the overall results, we found:

- A high scatter in the results for all the categories. There has been no deep analysis into the cause of this yet, but the websites we have evaluated have certainly shown different degrees of maturity in the search functionalities they offer, which likely is one of the contributing factors.
- The median performance is 0.5 or lower, indicating that a lot of potential for improvement still exists for many of the IR applications we evaluated.
- The category “User Interface” has lowest mean and smallest standard deviation. This is somewhat surprising, given that there are well-known examples from the field of Web search services, which are good blueprints for what users expect from search functionalities today.
- “Search Result” is the only category where the maximum score is reached.

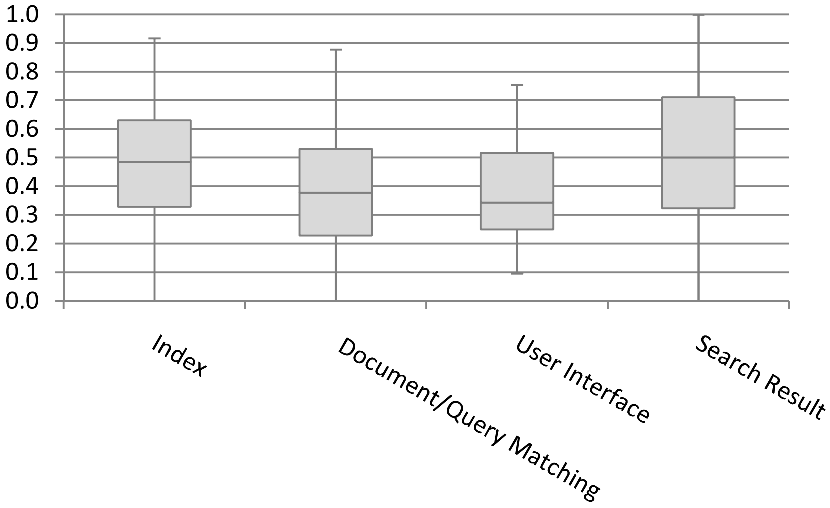

Fig. 3. Overall results from guerrilla campaign

Further exploration shows that no website scored consistently high for all categories. Again, this is testament to the potential for improvement of the underlying IR applications.

We can partition the results for the individual criteria in three groups (see Table 3):

- good results (generally the tests are passed by most of the sites)
- poor results (most of the sites failed these tests)
- neutral (no general tendency, some sites pass, some fail)

Some of the preliminary conclusions we draw from the observations during the guerilla campaign include:

For the category "Indexing", we found applications lacking in terms of "freshness". Operators should pay care to keeping the index fresh, i.e. to choose an appropriate interval for updates. Further, stemming is still not employed in many applications, and can help to boost both retrieval effectiveness and make handling more transparent for users.

For the category "Matching", few applications provide strong functionality for users to give feedback about the search results (and thus, ultimately, influence the search mechanism) – which seems counter to the idea of modern Web services involving the user more deeply. Multimedia retrieval has not found widespread adoption in the applications we tested so far.

For the category "User Interface", we found a lack of functionality for user guidance – i.e. tools such as term suggestion or spell checking components. Across all applications, functionalities that let users benefit from other users (such as display of related content or context information; or provisions for sharing results) are still not widely adopted.

Table 3. Criteria Results

GOOD RESULTS	NEUTRAL RESULTS	POOR RESULTS
— Completeness — Phrasal Queries — Performance/ Responsiveness — Browsing — Known Item Retrieval — Diversity	— Office Document Handling — Separation of Actual Content and Representations — Special Characters — Duplicate Documents — Meta Data Quality — Tokenization — Named Entities — Query Syntax — Over- and Under Specified Queries — Cross-Language IR — Exception Handling — Result List Presentation — Entertainment — Localization — Facets — Sorting of Result List — Justification of Results — Navigational Queries	— Freshness — Synonyms — Stemming — Feedback — Multimedia — User Guidance — Personalization — Social Aspects — Result List Import/ Export — Monitoring — System Override — Related Content — Context Information — Navigational Aids — Mobile Access — Geo-Location

The scores in the category “Search Result” were overall the best of the main categories. Still, ample opportunities for improvement remain, such as the inclusion of geo-location information into the ranking of results.

When looking at the scores that testers assigned for their subjective experience, we find a correlation of 0.53 between this “user experience” and the overall scores. This is an encouraging indication that scores derived from the evaluation methodology are actually useful estimates of user perception.

8 Outlook / Future Work

There are two logical next steps to improve the presented methodology. First and foremost, the limitations of the methodology should be more closely examined and remedied, if possible. More precisely, the scoring of individual tests is to be revisited. A scientific rationale for score ranges needs to be elaborated. As it stands, scoring has been designed to be very coarse to facilitate result aggregations and weighting. While the design was shown to be practical in the validation campaign, some tests might benefit from more granular scoring, according to estimated degrees of user satisfaction probabilities. Such estimations can be based in part on our work on best practices for information retrieval applications [11].

Another worthwhile effort is the description of test automation possibilities, including guides as to how automation of individual tests can be achieved. An envisioned result of that effort is a tool suite which can be used to instrument applications and run automated tests, providing an evaluation and monitoring tool for industry practitioners.

Acknowledgements. Most of the work reported in this paper has been supported by the PROMISE network of excellence[2] (contract n. 258191) project as a part of the 7th Framework Program of the European commission (FP7/2007-2013). The authors would like to acknowledge the contributions and support of all the different Promise partners, mainly to use case specific adaptations, criteria selection and conducting the validation of the methodology, among other points. See also [1] for a list of contributors to the methodology. The work is based on two earlier studies on the evaluation of the search functionality of enterprise web portals, see [12] and [13].

References

1. Rietberger, S., Imhof, M., Braschler, M., Berendsen, R., Järvelin, A., Hansen, P., García Seco de Herrera, A., Tsikrika, T., Lupu, M., Petras, V., Gäde, M., Kleineberg, M., Choukri, K.: PROMISE deliverable 4.2: Tutorial on Evaluation in the Wild (2012)
2. Robertson, S.E., Maron, M.E., Cooper, W.S.: Probability of relevance: a unification of two competing models for document retrieval. Info. Tech: R. and.D 1, 1–21 (1982)
3. Cleverdon, C.W.: The Cranfield tests on index language devices (1967)
4. Voorhees, E.M.: The philosophy of information retrieval evaluation. In: Peters, C., Braschler, M., Gonzalo, J., Kluck, M. (eds.) CLEF 2001. LNCS, vol. 2406, pp. 355–370. Springer, Heidelberg (2002)
5. Jansen, B.J.: Search log analysis: What it is, what's been done, how to do it (2006)
6. Blecic, D., Bangalore, N., Dorsch, J., Henderson, C., Koenig, M., Weller, A.: Using transaction log analysis to improve OPAC retrieval results (1998)
7. Kohavi, R., Henne, R., Sommerfield, D.: Practical Guide to Controlled Experiments on the Web: Listen to Your Customers not to the HiPPO (2007)
8. Radlinski, F., Kurup, M., Joachims, T.: How Does Clickthrough Data Reflect Retrieval Quality? (2008)
9. Dunlop, M.: Reflections on Mira: Interactive evaluation in information retrieval. J. Am. Soc. Inf. Sci. 51, 1269–1274 (2000)
10. Borlund, P.: User-centered evaluation of information retrieval systems. In: Information Retrieval: Searching in the 21st Century, pp. 21–37 (2009)
11. Braschler, M., Rietberger, S., Imhof, M., Järvelin, A., Hansen, P., Lupu, M., Gäde, M., Berendsen, R., García Seco de Herrera, A.: PROMISE deliverable 2.3: Best Practices Report (2012)
12. Braschler, M., Herget, J., Pfister, J., Schäuble, P., Steinbach, M., Stuker, J.: Evaluation der Suchfunktion von Schweizer Unternehmens-Websites (2006)
13. Braschler, M., Heuwing, B., Mandel, T., Womser-Hacker, C., Herget, J., Schäuble, P., Stuker, J.: Evaluation der Suchfunktion deutscher Unternehmens-Websites (2009)
14. Peters, C., Braschler, M., Clough, P.: Multilingual Information Retrieval: From Research to Practice. Springer (2012) ISBN 3642230075

[2] http://www.promise-noe.eu/