PROMISE

Participative Research labOratory for Multimedia and Multilingual Information Systems Evaluation

FP7 ICT 2009.4.3, Intelligent Information Management

Deliverable 2.3
Best Practices Report

Version 1.02, 31 August 2012

Network of Excellence co-funded by the 7th Framework Program of the European Commission, grant agreement no. 258191

Document Information

Deliverable number:	2.3
Deliverable title:	Best Practices Report
Delivery date:	31.08.12
Lead contractor for this deliverable	ZHAW
Author(s):	Martin Braschler, ZHAW Stefan Rietberger, ZHAW Melanie Imhof, ZHAW Anni Järvelin, SICS Preben Hansen, SICS Mihai Lupu, TUW Maria Gäde, UBER Richard Berendsen, UvA Alba Garcia Seco de Herrera (HES-SO)
Participant(s):	ZHAW, SICS, TUW, UBER, HES-SO, UvA
Workpackage:	2
Workpackage title:	Stakeholders Involvement and Technology Transfer
Workpackage leader:	SICS
Dissemination Level:	PU – Public
Version:	1.02
Keywords:	Best Practice, Information Retrieval Application

History of Versions

Version	Date	Status	Author (Partner)	Description/Approval Level
0.10	30.05.12	Draft	ZHAW	Initial version
0.20	26.06.12	Draft	ZHAW	Integrated partners comment
0.30	15.07.12	Draft	ZHAW	First complete version circulated to partners
1.00	17.08.12	Draft	ZHAW	Final version submitted for internal review.
1.02	31.08.12	Final	ZHAW	Final version after internal review

Abstract

This report presents best practice recommendations for information retrieval (IR) system developers, IR application implementers and IR application maintainers. It covers the main aspects of IR systems and applications, as well as recommendations for the user interface and evaluation. The best practices presented are the result of a distillation of academic IR output, taken mainly from experiments conducted within the confines of the CLEF evaluation campaigns, but also from additional sources. Elaboration was carried out both as a manual, intellectual effort, but also using semi-automatic, statistical methods that provided additional evidence for validation. Information retrieval technology is today used for very diverse purposes, supporting a range from "classical" search engines to applications such as topic detection or recommender systems. It is thus important to provide context to the individual recommendations. The report proposes a structure for the different best practice recommendations that states limitations and qualifications for different use case domains, and is prepared to include direct links to experiments and tested configurations in the future.

Table of Contents

Executive Summary

Gathering, sorting, evaluating and retrieving information is increasingly becoming crucial for professional (information) workers and also in every-day life. Consequently, information retrieval (IR) technology has been widely adopted in some fields, with some IR applications, such as Web search services becoming immensely popular. Academic research has started to branch into many aspects covering advanced IR issues such as the handling of multilingual and multimedia information, but there is as yet little take-up in operational systems for the proposed solutions. This report presents best practice recommendations for information retrieval (IR) system developers, IR application implementers and IR application maintainers. A focus of the work has been on clear, concise recommendations, which are listed in a structured form, accompanied by a more detailed discussion of associated aspects. In the following table, all recommendations from the report are summarized. When applying the recommendations in practice, an appropriate subset needs to be selected first based on the context and use case domain of the IR application.

BP Title	Action
Retrieval Paradigm	IR system that underlies the IR application should support ranked retrieval (term weighting)
Character encoding	Use Unicode for all text encoding
Document encoding	Use XML to encode data collection
Character normalization	Normalize diacritical characters to basic character representations. Convert characters to lowercase.
Tokenization/Business Entities	Domain specific terms containing typical tokenization characters (e.g. “-”, “/”, “:”, etc.) should be treated separately. Core business entities should be indexed as single features, where appropriate. If multilingual retrieval is offered, translation of business entities needs to be taken into account as well.
Stopword Elimination	Avoid stopword elimination. If not possible, use minimal stopword elimination. Choose weighting scheme that is robust with respect to stopword elimination
Stemming	Implement stemming
Decompounding	Implement decompounding component.
Character n-grams	Use character n-grams for indexing and retrieval
Matching	Use well-known, stable weighting schemes, such asLnu.ltn, BM.25, or Divergence from randomness
Recall	Use pseudo relevance-feedback to enhance recall.
Index Completeness	Make sure that all documents are reachable and processable by the indexer. Assign sufficient access rights and implement document processors for every type of document within the application.
Index Freshness	Update the index at least daily. Depending on the used weighting scheme and application architecture, partial index

	updates may be possible and in that case should be done.
Separation of Actual Content and Document Representations	Detect and remove structural document parts (e.g. headers and footers) before indexing. These parts do not contain actual document content.
Detect and Remove Duplicate Documents	Detect and remove duplicate documents when indexing using checksum or full document vector comparison.
Vocabulary Coverage (Translation Resources)	Maximize vocabulary coverage of translation resources. Add domain-specific resources.
Translation operation	Use document translation where possible. When the only textual description of the items is metadata, use translated metadata.
Translation robustness	Use combinations of translation resources. Also translate metadata, if available.
Interlingua	Use interlingua with care (where unavoidable)
Improve Meta Data Quality	Process all available meta data on documents. Enforce meta data curation on document entry into application.
Text-based Multimedia retrieval	Use textual retrieval when possible (i.e., if captions are available, or if speech can be transcribed)
Content-based multimedia retrieval	Use content-based retrieval when possible
Hybrid multimedia retrieval	Use content-based retrieval to refine results from textual search when possible
Document snippets	Offer document snippets (query-biased summaries) in the result list
Multilingual document summaries	Offer translated document summaries, containing the most important noun phrases, relevant passages, and key concepts.
(Language-independent) metadata	Use (language-independent) metadata as fall-back from full text translation
Result list sorting	Offer multiple ways to sort the result list; adapt to requirements of use case domain
Avoid user involvement in query refinement	Implement self-contained query refinement (e.g., blind relevance feedback) instead of interactive refinement techniques when potential for user confusion exists (e.g., if user does not understand the document language)
Provide additional context for matching items	Link additional sources and resources that provide context information (e.g., encyclopaedic content, maps, information on named entities...)
Improving ranked lists	Prefer listwise learning to rank over pairwise and pointwise learning to rank.
Deriving features	Derive features from several retrieval algorithms.
Feature selection	Use features that are quality indicators of documents, e.g., probability of being spam (spamminess), PageRank,

	freshness and so on. Use any metadata associated with documents, e.g., if there is authorship information, additional features may be derived such as reputation of author, credibility, and so on.
Feature normalization	Normalize features before feeding values into machine learning algorithms.

1 Introduction

Gathering, sorting, evaluating and retrieving information is increasingly becoming crucial for professional (information) workers, but also in every-day life. Since at least the 1950s, information retrieval (IR) as a discipline has studied how to find relevant information in response to requests formulated by users. While information retrieval technology has been initially adopted mainly in the library field, advances in processing power and storage capacity has seen increased introduction of enterprise retrieval systems in the 1990s. The wider public has been exposed to information retrieval technology through the introduction of Web search services since the late 1990s, with services such as AltaVista, Lycos, and ultimately Google, becoming available. While using information retrieval technology has thus become common-place for search and exploitation of *textual* content, there are still few operational systems for multilingual and multimedia information access today.

The increasing availability of content in many different languages and media, and the consequent need for global corporations to effectively process this information leads us to the belief that there is ample need for knowledge transfer between the academic information retrieval community on the one hand, and IR system developers and IR application implementers and maintainers on the other hand, in order to accelerate the rate of technology take-up.

While the academic community has increasingly focussed on multilingual and multimedia information retrieval and access, with campaigns such as TREC (http://trec.nist.gov) and CLEF (http://www.clef-initiative.eu) devoting tracks and tasks to these issues, it is difficult for practitioners to access this academic "output". Firstly, experiments are often conducted at the system level, i.e., mainly concerned with the effectiveness of retrieval algorithms in the narrower sense, and not so much in how information retrieval supports a larger software application in an operational setting. Secondly, and maybe even more importantly, many hundreds of papers need to be sighted, weighed against each other, and distilled into concrete action for implementation or deployment. One of the main problems when thus "digesting" the academic output is assessing the validity of an academic experiment's results when compared to a specific operational setting. While information retrieval has a strong heritage of experimental evaluation, and descriptions of retrieval experiments are abundant, each experiment has a very specific setting, including restrictions and limitations that derive from that setting. These are not necessarily clearly stated in the paper (they may be noted in the track guidelines and other places), and generalizing across different experiments is thus hard without being deeply immersed in IR literature.

The strong emphasis on use case domains in Promise makes it worthwhile to revisit some assumptions about IR systems and applications in the confines of the Promise network, and a task on the distillation of best practices was thus introduced into the Promise work programme. In contrast to some earlier attempts at IR best practices distillation (see, e.g.,

[Braschler & Gonzalo 2009]), Promise will also strive to extend its infrastructure in terms of the DIRECT system in order to be able to actively curate the resulting best practice recommendations.

Not least, having a comprehensive set of best practices provides a better understanding of what an IR application is (vs. an IR system), and how it should be evaluated. We hope it leads to the possibility to better describe the right setup of IR applications.

It should be noted that while one of the intents of compiling the best practice recommendations is to provide advice on IR system deployment or IR application implementation that is as broadly applicable as possible, the validity of some of the recommendations is by necessity limited to only some use case domains. We have tried to note where recommendations are assumed to have such limited validity, but for very specialized use case domains, which are substantially different from basic information retrieval or from the use case domains covered by Promise, additional considerations may be necessary. Also, while best practice recommendations provide helpful guidelines for assessing an IR application, they do not provide a substitute for IR system or IR application evaluation, where usually a quantitative analysis of aspects including effectiveness, usability and others is made.

The breadth of tasks and business processes which can be supported by IR systems and applications is great; and it is thus by necessity that a project such as Promise and a report such as the present cannot cover all possible aspects for all possible use case domains. We hope that the present report is a helpful summarization of best practices that address a substantial part of many IR applications. A lot of work remains for future improvements: a main extension is planned in linking the best practices to sets of concrete experiments from the CLEF campaign, thus giving the reader a very powerful tool to quickly access the relevant parts of IR literature. To this end, both experiments and best practices will have to be curated systematically, which is planned for future iterations of the DIRECT system. The template for best practice recommendations has already been extended to provide the necessary field ("tested configurations").

In the following, we define the term “best practice” (BP), as it is used in the context of this report:

> *“Methods and techniques that have consistently shown results superior than those achieved with other means, and which are used as benchmarks to strive for. There is, however, no practice that is best for everyone or in every situation, and no BP remains best for very long as people keep on finding better ways of doing things.”* [1]

While there are a multitude of definitions of “best practice”, these common points are evident:

1. BPs are bound to a set of conditions and circumstances to achieve the expected result.
2. BPs evolve or become obsolete over time due to technological or methodological advancements.

As defined above, this report aims to provide a set of best practices in IR for system implementers and application deployers and maintainers based on the state of the art of the field. We aim to summarize a set of best practices that generalizes across a large set of different IR applications.

[1] http://www.businessdictionary.com/definition/best-practice.html

We use an application-centred approach in defining our BP recommendations. The IR application in this context is any software solution which enables a user to work on an information access/retrieval scenario (e.g., one of the use cases domains as identified in the description of work of the Promise Network of Excellence). Figure 1 shows how an Information Retrieval application supports an information access cycle.

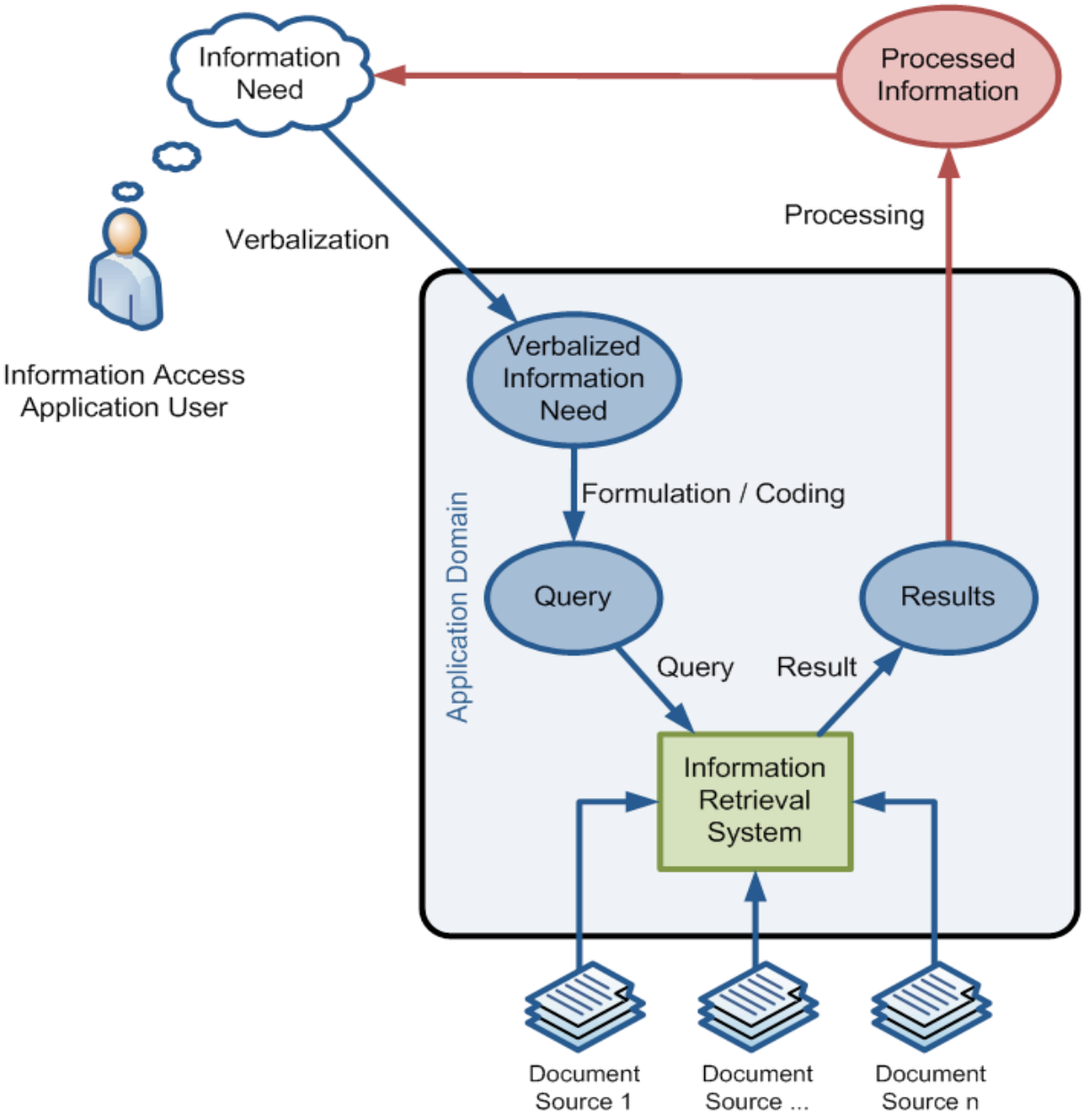

Figure 1: Information Access Cycle with Application Domain

When talking about an IR application, one thus needs to consider an *entire* set of IR system, the document collection, and the specific configuration. An IR application supports the

entire information access cycle. Contrast this with many academic IR evaluation experiments, where mainly the IR system proper, i.e., the component that matches coded queries with document representations, is evaluated. The popular Cranfield paradigm is most often used in this scenario, and is well understood and well suited for these types of IR system (component) evaluation [Cleverdon 1967] [Voorhees 2002]. However, as can be seen in Figure 1, for operational IR applications, this view is too narrow, and we have tried to address this in the report by giving best practice recommendations for all aspects of a multilingual and multimedia IR application.

A BP's fitness for a particular purpose is given by its validity description. However, some scenarios call for specific components and thus BPs concerning those components are only ever valid in the context of these scenarios.

The report covers the following areas:

- System (Indexing, Translation, Matching) and Application (e.g., Use Case Domain)
- End user[2] interface (e.g., Personalization)
- Evaluation (e.g., how to derive better configurations)

2 Formalization

2.1 Structural Elements for Best Practice Recommendations

The present report tries to balance the need for very careful description in academic literature with the desire for maximum clarity by practitioners. A simple "Recommendation → Based on" scheme containing an unambiguous description was deemed a suitable approach and initially has been tested in [Braschler & Gonzalo 2009]. In Promise, we have aimed to substantially refine and improve on this scheme, by adding more structure that allows giving richer information, among other things regarding the applicability to different use case domains. This is a major contribution of this report.

To this end, we introduce the following structural elements, where applicable:

1. **Validity and Qualification**: Restrictions of applicability depending on use cases domains or other circumstances.
2. **Dependencies**: Dependencies and effects on other BPs.
3. **Action**: Concrete action plan: e.g., features to be implemented or steps to be taken when following the BP.
4. **Expected Impact**: Quantified (if possible) impact of the implementation of this BP on the application, the system or on individual components.
5. **Support of Claim**: Source material for BP, including references, experiments, etc.
6. **Tested Configurations**: Actually implemented and tested configurations of the BP which performed well. Currently, this is mostly a placeholder. In the future, it is desirable and foreseeable that rich sets of experiments can be linked for each BP. Ultimately; external contributors should be able to add to the element through the DIRECT system.

The descriptive text should contain at least "action", "impact" and "support of claim". The

[2] "End user" meaning a user of an actual retrieval application

order is based on triage criteria. Readers shall be able to decide quickly whether a BP is relevant for them. They shall first be confronted with statements about the applicability of the BP in their current context. Afterwards, dependencies on other BPs, if any, shall be described.

2.2 Background on Structural Elements

Validity and Qualification

Through the use of this element we reflect the focus of Promise on concrete use cases domains.

Dependencies

If any dependencies on other BPs are known, they are described here. This includes both beneficial and detrimental effects, as well as full reliance on or mutual exclusion of particular other BPs.

Action

The BP must contain clear instructions to follow.

Expected Impact

While the impact cannot always be quantified, an estimate is given when applicable. Implementers shall decide if the expected impact fits the requirements of their own application. Some of the best practices quantify the expected impact in terms of a change in retrieval effectiveness. This is usually measure by using the measures precision (ratio of relevant retrieved items to all retrieved items) and recall (ratio of relevant retrieved items to all relevant items). Both measures can be combined into a single measure "mean average precision" (MAP) that is indicative of performance across different recall levels and different queries. If robustness is a focus, "geometric mean average precision" (GMAP) is used. For details on how to calculate and interpret these measures, we refer the reader to IR literature, such as [Peters et al. 2012].

Support of Claim

This element shall provide the evidence that the instructions as formulated in the “Action” element are preferable to other comparable methods and affect an application as described in “Expected Impact”.

Tested Configurations

While the “Action” element clearly states *what* should be done, it may omit *how* it is supposed to be done. Parameter values, technologies, component setups, etc. can be configured for different purposes and scenarios. Actually tested configurations and experience recounts will be recorded in this element once this is supported by the evaluation infrastructure.

3 Distillation and Elaboration

The elaboration of best practices is a product of human intellectual effort and evaluation. A first source to discover any preferable approach are directed experiments, providing empirical evidence as to which approach is best under which circumstances. Another source is scientific consensus among experts. We have employed statistical and IR methods as tools to facilitate the identification of evidence and consensus.

To provide a validation for the recommendations given in this report, we performed a

statistical analysis of the Cross-Language Evaluation Forum (CLEF) experiment descriptions. By this procedure, we intend to minimize the risk that core concepts are omitted during the intellectual elaboration process for the best practices. In a sense, our statistical analysis provides an "ultra-dense" summary: the body of literature is condensed into lists of statistically characteristic phrases. The unstructured text of the CLEF working notes is parsed to build an index using the information retrieval system *Lucene*[3] (the procedure does not use any special Lucene features, and could be implemented with a number of alternative IR systems). We assume that characteristic phrases are statistically frequent in the collection. The resulting lists can then be used as queries for interactive searching in the collection of experiment descriptions, thus greatly aiding the manual BP elaboration process.

Our first approach is to approximate the linguistic concept of a phrase by extracting two-word collocations based on their document frequency (df). In practice, the text is split into a sequence of overlapping pairs of content-bearing terms (for the distinction between content-bearing and non content-bearing terms, see the discussion of stopwords below). The document frequency is a measure of the number of documents that contain a phrase. We thus get an indication which phrases are globally frequent in the collection. We further use the collection frequency (cf) of a phrase, which represents the total number of occurrences of the phrase in the collection (summed over all documents). In contrast to the document frequency it is possible that a phrase has a relatively high collection frequency even though it appears only in a few documents (local "hotspots" of phrase use).

In total, 1121 experiment descriptions of CLEF from 2000 to 2011 have been analyzed. Statistics such as df and cf get influenced by the skewed distribution of term frequencies in natural language text. In particular, certain "non-content-bearing" words, such as articles, particles, interjections etc. (commonly referred to as "stopwords") tend to dominate these statistics. Phrases containing these words can thus "flood" the resulting lists of phrases. To counter this effect, stopwords have been removed based on a list of stopwords supplied with the Terrier[4] system. When dealing with academic literature or more specifically with experiment descriptions in the form of academic papers, there are furthermore a significant number of phrases that are characteristic to this form of text, but not so much for the content of the text (i.e., the experiment description). We have thus manually compiled a list of "stop-phrases" (e.g., "following section", "future work" etc.) and removed those from the text as well. By applying stemming, i.e., an algorithmic reduction of word forms to a "stem" that omits inflectional and derivational suffixes (and thus allows conflation of related word forms), we avoided to have the same concepts appear multiple times in the lists (e.g., in singular and plural form). Table 1 lists the top-ranked phrases in their "stemmed" form with the document frequency and the collection frequency (cf).

Term	df	cf
averag precis	446	1860
relev document	407	1603
document collect	396	1266
queri expans	372	2053
search engin	362	1058
queri term	329	1157

3 http://lucene.apache.org/

4 http://terrier.org/

natur languag	325	654
relev feedback	306	1245
document retriev	288	745

Table 1 The top-ranked two-word phrases extracted from the CLEF experiment descriptions. The ranking is based on the document frequency and the phrases are shown in their "stemmed" form.

Most of the frequently used phrases thus extracted are indeed referring to concepts that are essential in the information retrieval domain. We thus feel that the list can give a rough overview over the different concepts employed in the experiments. Most importantly, it serves for validation that no important concepts have been *overlooked* by the elaboration process.

However, it can be argued that these "highly frequent" concepts are those that are least likely to be missed. It is thus also interesting to identify topics that are specific to a subgroup of the analyzed working notes (such as a "cluster" of topically similar experiment descriptions). Therefore, we performed a second statistical analysis based on reduced representations of the experiment descriptions. For each paper, we compile a list of its *n* most discriminative terms. We consider a term to be discriminative when it occurs frequently within some of the documents (locally frequent), but rarely in the collection as a whole (globally frequent). These considerations underlie the well-known "term frequency-inverse document frequency" (tf-idf) score, which was thus used for this task. The reduced representations describe the content of the documents in the context of the examined collection. We loaded the representations into a Lucene index and extracted once more the top-ranked two-word phrases based on their document frequency. Table 2 shows the ten top-ranked phrases from the CLEF working notes using an automatically generated representation of *30* terms. The document frequencies as well as the collection frequencies are smaller than in Table 1 since we calculated the frequencies based on the representations instead of the raw text of the working notes. As intended, the extracted terms reflect topics which are very much at the core of some experiments, such as answer extraction and plagiarism detection, but which are not addressed by the large majority of experiments, as their applicability is limited to some tasks.

Term	df	cf
answer extract	27	27
plagiar detect	26	27
patent document	23	23
queri queri	23	26
answer type	21	22
prior art	21	21
annot task	20	20
suspici document	20	21
visual concept	20	21

Table 2 The top-ranked two-word phrases extracted from the reduced representations. The ranking is based on the document frequency and the phrases are shown in their "stemmed" form.

4 Sources for Best Practice Recommendations

4.1 Sources and curation

During gathering, we must aim to describe only correct and complete BPs and find as many correct BPs as possible. To use an information retrieval analogy: we want to optimize both precision and recall of BPs.

To ensure that BPs are correct, several sources need to support the claim of any BPs. We have previously defined BPs as mutable concepts and consequently they must be continuously evaluated and validated after having been established.

Besides creating BPs through human effort alone, there are statistical methods which can support the creation process. Aside from the phrase analysis described above, metadata can be used for this purpose. If metadata for each system, component, experiment, etc. is stored, emergent BPs can be found through analysis of the metadata. This requires the Promise evaluation infrastructure to support annotations and storage of such kind of metadata. Furthermore, metadata must be accessible by means of a structured query language for analysis purposes. The DIRECT system will be extended in the future to make this kind of access to best practices possible.

Lastly, we need to consider the curation of obsolete BPs. We propose to retain obsolete BPs for historical reference and as a reminder of which new developments or insights have made them obsolete. Whenever a new BP replaces an old one, they should be linked by pointers. This is planned to be addressed by extending the DIRECT system to include best practice recommendations as an entity of its underlying database.

4.2 Exploitation and validation

We see potential in formalizing compliance levels for components and systems. As an example, assume that eight BPs have been established for the area of document indexing. Someone starts an experiment and annotates the used components accordingly. Six out of eight known BPs are implemented which yields an indexing component compliance level of 75%.

Whenever a full system or an indexing component is used in an evaluation on the Promise infrastructure, that system's or component's compliance with the established BPs should be identified. For simplicity's sake, we shall define that any BP can only be implemented

fully or not at all.

Automatic measurements should be implemented for BPs which support it, allowing for a low-effort correlation of compliance levels to system/component performance. This would help to validate established BPs either on their own or in conjunction with other BPs. This in turn enables us to investigate the effect of interdependencies between BPs on full system performance.

5 Detailed Best Practices Descriptions

5.1 Template

Best practices recommendations are presented using the following template (see descriptions for the individual fields for an explanation of their use). Additionally, we have attempted to provide more context by adding a discussion of associated aspects for each recommendation.

[BP Template]
Validity and Qualification
Restrictions of applicability depending on scenarios / use cases or other circumstances
Dependencies
Dependencies and effects on other BPs
Action
Detailed action plan of features to be implemented or steps to be taken when following the BP.
Expected Impact
Quantified impact of the implementation of this BP on the system or component it is being implemented into.
Support of Claim
Source material for BP, including references, experiments, etc.
Tested Configurations
Actually implemented and tested configurations of the BP which performed well

5.2 System, Application

These are best practice recommendations that directly address the "system proper", i.e., the Information Retrieval system in the narrow sense: a system that accepts an encoded query (most often a character string, but potentially in any media) and returns a set of items that best match that query. For a comparatively long time already (see, e.g., [Robertson 1977]), and even more so over the last decade with increasing influence of Web search services on daily life, the prevailing paradigm for structuring that set of items has been to return a ranked list of items, typically ordered by the estimated probability of the relevance of the item with regard to the query. This retrieval paradigm of using "ranked retrieval" has replaced using Boolean expressions ("Boolean retrieval") in many cases. Note, however, that, as [Kim et al. 2011] point out, Boolean search remains very popular for some professional applications. However, they go on to note that "This is not because Boolean

queries are the most effective. In fact, a number of studies over the years [...] have shown that “keyword” queries are often significantly more effective.". Boolean querying functionality may be indispensable in some environments where the "predictability" of retrieving a specific set is important. Having a Boolean expression allows easy interpretation of why an item is included or excluded in the set (addresses issues of auditability, for instance). In other cases they are mainly popular for historic reasons [Azzopardi et al. 2010]. In such cases, we urge to explore at least the possibility to offer users a choice between the paradigms.

In ranked retrieval, the focus on "topical" relevance has been recently scrutinized more closely, and alternative proposals for worthwhile criteria, such as diversity, novelty, robustness etc. have been made.

By considering "novelty", the focus is on an avoidance of redundancy [Clarke et al. 2008]. Ideally, the document ordering for a query would allow the user to gain new knowledge with each document that is considered. Lately novelty is often evaluated in conjunction with "diversity". Citing [Clarke et al. 2008], "diversity" refers to the need to resolve ambiguity. The goal is to return a diverse ranked list that covers all the meanings of a query, while avoiding excessive redundancy (linking back to the concept of novelty).

For the time being, this broad consensus with respect to superior retrieval effectiveness using a retrieval paradigm of "ranked retrieval" leads to the following best practice recommendation:

Retrieval Paradigm
Validity and Qualification
Systems that support unstructured search on potentially heterogeneous information, on information from many sources. Information needs are vague and/or not always well-formed. There may be use case domains where special considerations (auditability etc.) mandate the use of different retrieval paradigms.
Dependencies
Action
IR system that underlies the IR application should support ranked retrieval (term weighting)
Expected Impact
Better retrieval results than systems using classical Boolean retrieval
Support of Claim
Nearly universal use of systems based on ranked retrieval in the CLEF, TREC and NTCIR evaluation campaigns. Exceptions are rare, and either very limited in their application, or did not yield competitive results (see, e.g., [Ripplinger 2000])
Tested Configurations
Too many to list. Encompasses all different tracks/tasks covered by the CLEF, TREC and NTCIR evaluation campaigns.

None of the classical retrieval mechanisms, such as the vector space model [Salton et al. 1975] or probabilistic weighting schemes [Robertson et al. 1980] have any inherent

dependencies to character encoding schemes. They are broadly applicable to most schemes, as essentially, they work on byte string (or even numeric) representations of the indexing features. However, this independence from specific character encodings should not be used as an excuse to use legacy encodings. Indeed, freedom in the choice of encodings implies that the most universal character encoding should be used.

Character encoding
Validity and Qualification
Multilingual systems
Dependencies
Action
Use Unicode for all text encoding
Expected Impact
Smaller overhead for handling documents in many languages
Support of Claim
Simplifies handling of source documents without adverse impact on retrieval effectiveness
Tested Configurations
Not directly tested in IR evaluation

Similar to character encoding schemes, none of the classical retrieval mechanisms have any inherent dependencies to document encoding schemes. Again, they are broadly applicable to most schemes, as essentially, they make very little use of document structural information. An example to the contrary is weighting approaches to take account of hyperlink structure, such as HITS [Kleinberg 1999] and Page Rank [Brin & Page 1998]. However, these approaches have thus far been found to be only applicable to very large document collections, such as the collections collected from the World Wide Web by Web search services. This independence from specific document encodings allows the use of flexible representations. All large IR evaluation campaigns today deliver their data mainly in XML form, and XML encoded documents should be supported by nearly all IR systems underlying IR applications today (examples include the popular open-source IR systems Terrier and Lucene).

Document encoding
Validity and Qualification
Use in systems for ranked retrieval (see below; XML can encode all information necessary for IR systems based on ranked retrieval)
Dependencies
Retrieval paradigm; Matching
Action
Use XML to encode data collection
Expected Impact
None on retrieval effectiveness; but should allow easy interoperation with other software components.
Support of Claim
See delivery mode of IR evaluation campaigns, see, e.g., open source IR systems such as Terrier and Lucene
Tested Configurations

The indexing process of an IR system dictates how retrieval items are represented in the system for matching. It is on this indexed representation that matches need to be found – if the indexing process is not appropriately configured, there is a danger of erroneous extra matches (leading to lower precision) and missed relevant matches (leading to lower recall). The fact that the items (often documents) stored in the IR system are typically not curated by the searcher (i.e., their contents are unknown to the searcher) coupled with uncertainty by the searcher on the items' content (users have an information need that is typically hard to express properly without already knowing the "solution"), makes it important that the indexed representation is in a normalised form that can guarantee matches across a variety of different formulations of the same information. To this end, in modern IR systems the items are processed in various steps, leading to a bag (multiset) of features representation of the item (see [Peters et al. 2012]). Features can be of any suitable type – for textual items (documents) normalised character strings ("stems") are typically used, and multimedia content can be represented in various ways, e.g., by using color information, brightness information, phonemes, information on edges etc. as features.

In retrieval of textual content, inconsistencies in the use of diacritical characters and capitalization of characters can be a hindrance to effective matching. Consider the Western European languages, for example, where words are capitalized when used at the start of a sentence, even if they are otherwise written consistently in lowercase. While capitalization may change the meaning of a word (such as "bush", a shrub, versus "Bush", a person name), and is thus potentially valuable in distinguishing different meanings of homonyms, such differing use, and a tendency by many users to enter search terms in lowercase to speed up typing, make discarding all capitalization information the most robust option. Similar considerations apply to the use of diacritical characters: while discarding diacritical marks may lead to the conflation of words with different meanings, this problem with extra matches has been found to be offset by the gain in correct matches that would have been missed due to inconsistent use (see, e.g., [Lazarinis et al. 2008] for a discussion of (non-)use

of diacritics by users of Web search services).

Character normalization
Validity and Qualification
IR applications/systems for textual content
Dependencies
Retrieval paradigm; Matching
Action
Normalize diacritical characters to basic character representations. Convert characters to lowercase.
Expected Impact
Better robustness. No significant impact on retrieval effectiveness based on average precision
Support of Claim
[McNamee & Mayfield 2003]
Tested Configurations

Enterprise IR systems and applications need to be finely configured to deal with the core entities of the business processes they support. Often, such entities are names, or multi-word terms. They also frequently contain special characters, such as hyphens. The systems need to be carefully adapted to index these entities the correct way. Examples include: "O'Brian", "F/A-18", "Coca-Cola", etc.

Tokenization/Business Entities
Validity and Qualification
Dependencies
Action
Domain specific terms containing typical tokenization characters (e.g., “-”, “/”, “:”, etc.) should be treated separately. Core business entities should be indexed as single features, where appropriate. If multilingual retrieval is offered, translation of business entities needs to be taken into account as well.
Expected Impact
Increase in robustness. Slight increase in effectiveness
Support of Claim
[Barcala et al. 2002][Pfeifer et al. 1996]
Tested Configurations

Retrieval effectiveness can be increased in some cases if only a subset of the features is retained. A major example is retrieval on textual content: due to the skewed distribution of word frequencies in text, with very few words being very frequent and the majority of words being very infrequent (so-called "Zipfian distribution [Zipf 1949]), the most frequent words often have a disproportionate influence on some weighting schemes. Unfortunately, such highly frequent words typically carry little information, being pronouns, particles, articles etc. IR literature denotes these as "non content-bearing words" or "stopwords". Their removal can increase retrieval effectiveness for those weighting schemes that are sensitive to this issue. Typical stopwords include in English: the, a, and, this etc. Their removal is not without problems, however, as any removal of features means that these retrieval items can no longer be located through these features. Consider, e.g., the rock band "The Who" – their name consists of two words that are typically eliminated as stopwords, thus compromising the ability to locate information on that band.

There are recent studies that indicate that the issues with weighting schemes malfunctioning when weighting stopwords no longer apply to some modern weighting schemes [Dolamic & Savoy 2009], [Becks et al. 2009]. Additionally, [Runeson et al. 2007] provide evidence that smaller stopword lists can outperform larger ones. In the interest of robustness, we propose to use these weighting schemes preferentially.

Stopword Elimination
Validity and Qualification
Retrieval on text. General retrieval (news texts and other domains); patent retrieval
Dependencies
Retrieval paradigm;
Action
Avoid stopword elimination. If not possible, use minimal stopword elimination. Choose weighting scheme that is robust with respect to stopword elimination
Expected Impact
Increases robustness of system. Increases total recall. Avoids problems with missing matches due to inadvertent removal of important information. No negative impact on retrieval effectiveness.
Support of Claim
[Dolamic & Savoy 2009], [Becks et al. 2009], [Runeson et al. 2007]
Tested Configurations

Retrieval items that consist mainly of natural language text may fail to be retrieved due to mismatches between the word forms used in the user's input and those used in the text itself. While in English, the number of word forms is typically small, some languages have a rich morphology (examples include Finnish, Turkish and others). For these languages it is imperative that the matching process is extended to match across different word forms. The components associated with this issue in IR systems are typically called stemmers: the character strings of surface words (i.e., the different word forms) are mapped to normalised, typically truncated "stems". Morphologically related surface words should be mapped to the same stem, although in rule-based stemmers errors can and do occur. Other than enabling a match in the first place, this conflation of word forms also aids with better weighting during the matching, thus typically boosting both precision and recall. Stemming has been shown to be beneficial across many languages (see, e.g., [Braschler & Ripplinger 2004]). When languages have a comparatively simple morphology, such as English, the removal of plural forms of nouns may be sufficient (see [Harman 1991]).

Stemming
Validity and Qualification
Dependencies
Retrieval paradigm; Matching
Action
Implement stemming
Expected Impact
Depends on language, small for English, larger (up to +30% average precision) for many European languages, even larger for languages with very rich morphology
Support of Claim
[Braschler & Ripplinger 2004], [Perez et al. 2009], [Harman 1991], [Hull 1996], [El-Khair 2007]
Tested Configurations

Tightly coupled with the issue of stemming is a linguistic phenomenon that, while infrequently observed in English, is prominent for some languages, such as German, Dutch or Korean: nouns can be joined to form "compound nouns". This compounding process can be very productive: in German, for example, compounds containing many different nouns can be formed (e.g., "Fussballweltmeister" – football world champion). While some compound words today are lexicalized, and are always written and used in their compound form (an example in English is "football"), many can be alternatively written in phrasal form ("Weltmeister im Fussball" – world champion in football). For matching, it is then essential that partial matches on the compound words are enabled – this is typically done using a decompounding component that splits the compound words into their constituents.

Decompounding
Validity and Qualification
Applies only to applications serving languages which contain many compound words (e.g., German, Dutch, Korean...). May also apply if complicated technical terminology is used frequently.
Dependencies
Retrieval paradigm; Matching
Action
Implement decompounding component.
Expected Impact
Important for total recall. Up to +30% average precision for some languages.
Support of Claim
German: [Braschler 2004]
Tested Configurations

Both stemmer and decompounding components typically incorporate at least some amount of linguistic knowledge which is language-specific. This makes deployment of such components costly in cases where many different languages need to be handled by an IR application simultaneously, and impossible in cases when such resources are not available for languages with a very limited speaker population. Studies, such as [McNamee & Mayfield 2004] and [McNamee & Mayfield 2003] have shown that character n-grams can be a useful substitute in such cases. When indexing character n-grams, all surface words are split into overlapping sequences of n characters, where n is usually set in the range of 3 or 5. Consider, for example, the word "football", which would be split into the character n-grams "foot", "ootb", "otba", "tbal", and "ball" for n=4. As can be seen, the technique leads to some effects that are similar to decompounding (the n-grams "foot" and "ball" cover only one constituent of the compound word each), and also guarantee at least some matches on n-gram level if different word forms are used, thus leading to a "stemming" effect as well. Notable drawback of the technique is the multiplication of the number of features to be indexed, which will lead to greatly increased size of the underlying data structures for matching (the index).

Character n-grams
Validity and Qualification
Systems dealing with languages for which little linguistic resources are available. Systems dealing with many languages simultaneously.
Dependencies
Retrieval paradigm; Matching
Action
Use character n-grams for indexing and retrieval
Expected Impact
Comparable performance to stemming
Support of Claim
[McNamee & Mayfield 2004], [McNamee & Mayfield 2003]
Tested Configurations

As mentioned earlier, Information Retrieval addresses the problem of matching a potentially incompletely verbalised information need with many different possible phrasings of information that is topically relevant to that need. As demonstrated, normalisation of the information is necessary to allow later matching. Even so, information retrieval by design cannot rely on exact match strategies as they are employed in database scenarios: the users should be allowed to formulate as much as possible about their information needs to increase the likelihood of matches, even when it cannot be realistically expected that relevant items would contain all those features. Further, the potentially non-optimal verbalisation of the information need means that not all search terms should be given equal weight. In practice, best match strategies are used, by using so-called weighting schemes that assign a score to every retrievable item with respect to the query. Note that this strategy is all the more necessary for multimedia queries: if querying the index with audiovisual features, typically not exact matches are sought, but matches with features that express the desired information (e.g., query by example for images: one concrete picture of a sunset may serve as a query to retrieve many other pictures of sunsets). The score assigned by the weighting schemes expresses a "similarity" (vector-space model) or a "probability of relevance" (probabilistic weighting schemes) – items are ranked according to the score, and presented in a list sorted by their rank.

The large IR evaluation campaigns TREC and CLEF have shown for many different IR-related tasks that some weighting schemes consistently perform well. We suggest using one of these weighting schemes unless precluded by some special requirements. Well performing weighting schemes, according to metrics such as mean average precision, include Lnu.ltn [Singhal et al. 1996], BM.25 [Walker et al. 1998], Divergence from randomness [Amati & Rijsbergen 2002], and others.

Matching
Validity and Qualification
Dependencies
Retrieval paradigm
Action
Use well-known, stable weighting schemes, such as Lnu.ltn, BM.25, or Divergence from randomness
Expected Impact
Delivers state of the art retrieval effectiveness across many different retrieval scenarios
Support of Claim
[Dolamic et al. 2008], many hundreds of experiments in the TREC and CLEF campaigns
Tested Configurations

Even when using normalised features to aid the matching process between the verbalised information need and the retrievable items, there will be items that are not found at all, even though they are topically relevant. Linguistic phenomena such as synonymy, metaphors etc. are among the main reasons for textual content, but similar effects can also be observed for multimedia content. Relevance feedback can help with this problem, by extracting additional search features from already discovered relevant content. The additional features are selected to be characteristic of the retrievable items they are derived from, specifically; they are selected to be frequently occurring in relevant content, while being (relatively) rare in the overall collection of retrievable items. To avoid having to determine relevant items for specific queries by hand, a variant called pseudo relevant feedback derives the additional features from highly-ranked items (after an initial search) instead. Using the technique, it is possible to match items with queries even when they originally do not contain any common features. As a drawback of the technique, not all queries benefit from this expansion with additional features. On average, small to moderate benefits in terms of mean average precision have been found for many different scenarios. These increases in effectiveness come at the expense of reduced efficiency: for every query, two retrieval passes need to be made (an initial pass to locate the highly ranked items, and a second pass after the extraction of the additional features).

If robustness is a pressing concern, the decrease in performance for some of the queries may be reason to avoid the technique. Also note that in some scenarios, receiving "hits" for items that do not share any direct matches with the search terms can be confusing to users.

Recall
Validity and Qualification
Recall-oriented retrieval scenarios (e.g., patent retrieval)
Dependencies
Retrieval paradigm; Matching
Action
Use pseudo relevance-feedback to enhance recall.
Expected Impact
Improved total recall. Slight improvement in mean average precision. Some queries may show decreased retrieval effectiveness.
Support of Claim
[Dolamic et al. 2008] [Moulinier & Williams 2005]
Tested Configurations

Indexing and matching work as a "tandem" to retrieve relevant items in response to the users' requests. It is however important to note that even the best indexing strategy and the best matching algorithm depend on a complete item collection: i.e., what is not part of the IR application's underlying collection of retrievable items, cannot be found using any strategy. It is therefore important that the IR application includes all potentially relevant retrievable items, and that the indexing strategy incorporates all content-bearing parts of these items in the index.

Index Completeness
Validity and Qualification
Dependencies
Action
Make sure that all documents are reachable and processable by the indexer. Assign sufficient access rights and implement document processors for every type of document within the application.
Expected Impact
Better robustness.
Support of Claim
Tested Configurations

A related issue to "index completeness" is "index freshness". A fresh index contains all

potentially relevant retrievable items, regardless of their recency. This means that new items are discovered and/or added within a reasonable timeframe, but also that updates to previously indexed items are carried out in a timely manner, and that obsolete items are deleted.

The latter two issues are typically non-trivial. Underlying most IR systems for ranked retrieval is a data structure called "inverted index", which allows extremely efficient calculation of best match scores for query/item pairs (essentially just one constant-time lookup per search term is needed). However, the implementation of the inverted index via hash tables is little suited to real-time updating: updates (inserts, deletes) are costly operations, in that they have to "touch" and lock entries in the hash tables that are associated with individual indexing features – and these entries are used for the calculation of retrieval scores. Updates are thus typically deferred – if the IR application in question is not used continuously around the clock, they can be scheduled at intervals of low use (e.g., night time). The interval scheduling needs to be suitable to the use case domain(s) addressed. The definition of "freshness" may vary considerably depending on the use case domain. While for some, such as the "cultural heritage" domain, updates may be infrequent, for some "social media" applications "freshness" may be defined in terms of seconds. In such cases, switching between multiple indexes or other measures (layered updates etc.) need to be considered.

Index Freshness
Validity and Qualification
For the Promise use case domain of Search for Innovation, freshness is defined by the application's claim. There may be other applications where the base collection is only rarely updated by design, such as the cultural heritage domain.
Dependencies
Requires Index Completeness
Action
Update the index at least daily. Depending on the used weighting scheme and application architecture, partial index updates may be possible and in that case should be done.
Expected Impact
Support of Claim
Tested Configurations

Standard weighting schemes used in IR systems, such as Lnu.ltn, BM25 and others (see "Matching") do not take document structure into account. They operate on an unordered multiset of indexing features. In practice, this means the phrases "man bites dog" and "dog bites man" are equivalent for matching. While these weighting schemes have been shown to be effective regardless of this limitation, this also means that it is important to remove any parts of the retrievable items that are not part of the content proper, i.e., headers, footers,

copyright notices, navigation elements etc. Not only does this avoid erroneous matches, it also avoids a pollution of the internal feature frequency statistics for the words occurring in these extra elements, which can affect ranking.

Separation of Actual Content and Document Representations
Validity and Qualification
Dependencies
Action
Detect and remove structural document parts (e.g., headers and footers) before indexing. These parts do not contain actual document content.
Expected Impact
Less erroneous matches on non-content bearing structural elements
Support of Claim
Tested Configurations

All the widely deployed weighting schemes mentioned earlier in this report calculate the retrieval score for an item mostly independently from the scores of other items (there is technically a small indirect dependency in that many weighting schemes use an idf – inverse document frequency – statistic for calculation of scores. This statistic changes according to the content of the system's document/item collection). In practice, this leads to exact duplicates of documents getting all the same score, and potentially flooding the top ranks of the result list. Analogously, very similar documents ("near duplicates") will obtain very similar scores, again potentially dominating parts of the result lists. In many cases, users will not want to see the duplicate information, instead preferring a result list that presents many different ("diverse") relevant results. In some cases, such as the cultural heritage use case domain, (near-)duplicate discovery can however be crucial, e.g., to compare the different editions of a work. Even in such cases, however, a functionality to detect the duplicates in the system, and then cluster them in the result list is considered helpful. In all cases where no strong preference by the user for obtaining (near-)duplicates can be assumed, duplicates can even be removed from the result list.

While the detection of exact duplicates can easily be implemented through a checksum mechanism, detection of near-duplicates is harder. Typical checksum mechanisms react very sensitively to small changes, generating very different checksums. [Broder 2000] presents a fingerprinting approach as an alternative, which uses a fixed-length representation of the items, that can capture similarity and allows much faster near-duplicate detection than a brute force approach (which would entail a comparison of every item with every other item – a quadratic effort).

Detect and Remove Duplicate Documents
Validity and Qualification
Depending on the use case domain (especially Cultural Heritage), near-duplicates can be important and must not be removed.
Dependencies
Action
Detect and remove duplicate documents when indexing using checksum or full document vector comparison.
Expected Impact
More diverse result lists
Support of Claim
Tested Configurations

The next three best practices address multilingual IR systems and applications, i.e., those systems that make items in multiple languages accessible. If the user is to be enabled to access the whole collection in his or her preferred language, some sort of cross-language information retrieval (CLIR) is necessary, i.e., items have to be matched to queries formulated even when the languages are different. Typically, discounting "exotic" solutions, some form of translation is necessary to achieve this -either of the item, the query, or both. While machine translation is the most widespread approach to document translation, the picture for query translation is not so clear. Queries are typically short, ungrammatical sequences of keywords, and thus ill-suited for machine translation. Various approaches to translate queries based on expansion and selecting the right translations from machine-readable dictionaries exist (see [Peters et al. 2012]).

Translation introduces a new source of ambiguity and error into the retrieval process. Inevitably, some queries will suffer: if key concepts cannot be translated due to failure of the translation resources, queries may return no relevant content at all. [Mandl et al. 2008] demonstrates that indeed cross-language information retrieval systems exhibit a much larger variance in retrieval effectiveness across different queries than monolingual ones. It is thus postulated that robustness is an important aspect of such multilingual IR systems and applications, and that the coverage of translation resources should be strived to be maximized, in order to cover as much specialized terminology as possible.

Vocabulary Coverage (Translation Resources)
Validity and Qualification
Dependencies
Retrieval paradigm; Matching
Action
Maximize vocabulary coverage of translation resources. Add domain-specific resources.
Expected Impact
Better recall. Better robustness.
Support of Claim
[Braschler & Gonzalo 2009]
Tested Configurations

IR weighting schemes typically use a limited set of corpus statistics to compute their scores. Among the most frequently used statistics are within-item feature frequency (how often does a feature occur in an item), document frequency (how many documents/items does a feature occur in), and some form of document/item length. When multiple languages are in play, some of these statistics get hard to maintain, as the same character stings can be meaningful words in multiple languages (with or without a shift of meaning. For example, "Paris" is refers to the same city in a number of languages, including French, English, and German. "Gift", on the other hand, denotes a present in English, but is a translation for "poison" in German). These statistics shouldn't be mixed up. As a consequence, scores computed for items in one language are not normally comparable to scores computed for items in other languages – a big problem if a single, multilingual result list is needed. This problem is also known as the merging problem ([Peters et al. 2012]) and can affect retrieval effectiveness significantly (a performance loss of up to 40% over a theoretical baseline that solves the problem has been observed).

The problem can be circumvented if the whole collection can be translated into a single language (document/item translation). This way, the index will be monolingual, and the statistics can be determined as in the monolingual case. As a side effect, the translated items may be suitable for presentation to the users in some cases.

If the replication of the collection through document translation is feasible, we advise to use this technique to avoid the merging problem. Please note, however, that many possible query languages could lead to many different translated item collections. To avoid this, an interlingua needs to be used. (see below).

Translation operation
Validity and Qualification
Local, centralized document collection. Limited number of languages.
Dependencies
Interlingua
Action
Use document translation where possible. When the only textual description of the items is metadata, use translated metadata.
Expected Impact
Avoid merging problem.
Support of Claim
Tested Configurations

As mentioned when discussing translation coverage, there is evidence that cross-language IR systems and applications exhibit greater variability in retrieval effectiveness across different queries than monolingual counterparts. Aside from increasing the coverage of single translation resources, a similar effect can be obtained by merging the output from different translation resources. Such a strategy may also be helpful in disambiguating translation output (see [Braschler 2004])

Translation robustness
Validity and Qualification
Dependencies
Action
Use combinations of translation resources. Also translate metadata, if available.
Expected Impact
Better total recall. Better robustness.
Support of Claim
Tested Configurations

The handling of a large number of different languages presents additional problems, regardless of a choice of document/item translation or query translation. In the case of query translation, the query must be translated into all different languages covered. In the case of document/item translation, the item needs to be translated into all possible query

languages. This issue can be avoided if an interlingua is chosen for matching: both the query and all items are translated into a common language; the matching process then is essentially a monolingual one. The main drawback of this method is the introduction of an additional translation step: essentially a match is the result of comparing two translated items, meaning that translation errors can multiply. However, a study by [Savoy & Dolamic 2009] gives evidence that using an interlingua does not necessarily lead to a decrease in retrieval effectiveness: when carefully choosing the interlingua so that the quality of translation resources is maximised, effectiveness can actually increase compared to a direct translation.

Interlingua
Validity and Qualification
System covering many languages simultaneously. Limited direct translation resources.
Dependencies
Action
Use interlingua with care (where unavoidable)
Expected Impact
Simpler implementation when many languages are handled at the expense of a slight decrease in retrieval effectiveness
Support of Claim
Tested Configurations

The frequency of occurrence of different queries in most cases follows a very skewed distribution: few queries will occur very frequently, while most queries will only occur once (for an example of this phenomenon see, e.g., [Silverstein et al. 1999], where a large query log of a Web search service was analysed). In enterprise IR applications, the frequent queries will be associated with core business entities aligned with the business processes underlying the application. If those entities are carefully curated using metadata, that metadata can be used to improve the retrieval quality of those queries.

Improve Meta Data Quality
Validity and Qualification
Applicability may be limited by issues of interoperability between different metadata standards
Dependencies
Action
Process all available meta data on documents. Enforce meta data curation on document entry into application.
Expected Impact
More robust retrieval results
Support of Claim
Tested Configurations

Multimedia retrieval based on text search requires a textual representation of the multimedia objects. For TV video streams this representation can be extracted from closed captions, teletext or subtitles. On web pages the words in the anchor text of a link to a multimedia object, the filename of the object itself, meta-data stored within the files or other context information such as captions can be used to represent the multimedia object as text. There are also methods to convert music to text, such as MIDI files. This can also be used for the query, which makes it possible to query by humming [Rüger 2009].

Text-based Multimedia retrieval
Validity and Qualification
IR applications that allow access to multimedia content (sound, video, still images,…). Valid for: web search (images, video, sound), music retrieval, video retrieval.
Dependencies
Content-based multimedia retrieval, hybrid multimedia retrieval
Action
Use textual retrieval when possible (i.e., if captions are available, or if speech can be transcribed)
Expected Impact
Support of Claim
[Rüger 2009]
Tested Configurations

Content-based methods are used successfully in medical image retrieval, since the data size is small and domain-specific features can be used [Li et al. 2009] [Kankanhalli & Rui 2007]. Furthermore, in the art and culture domain content-based methods are suited, since they can extract low-level features such as colours and textures, which are directly related to the query [Kankanhalli & Rui 2007]. Keep in mind, however, that formulating queries (audio-)visually can be difficult.

Content-based multimedia retrieval
Validity and Qualification
IR applications that allow access to multimedia content (sound, video, still images,…). Valid for: medical images, images in the art & culture domain
Dependencies
Text-based multimedia retrieval, hybrid multimedia retrieval
Action
Use content-based retrieval when possible (but compare to hybrid multimedia retrieval)
Expected Impact
Support of Claim
[Kankanhalli & Rui 2007] [Li et al. 2009]
Tested Configurations

The major techniques for multimedia information retrieval are text-based, content-based, and hybrid retrieval approaches. Hybrid multimedia retrieval tries to combine the retrieval results returned by text-based method and by content-based method to enhance the retrieval effectiveness. It was proposed to retrieve the result for a text query and then use a content-based refinement process. Empirical results show that hybrid retrieval is a promising method compared to the other two [Li et al. 2009].

In medical image retrieval at ImageCLEF the best approaches most often combine textual and visual features [Müller 2010].Also in the automatic search task at TRECVID a multimedia information need has to be satisfied in the best possible way without user interaction. In this task the best performing systems combine text retrieval with query classes, selection of detectors and query combinations [Snoek & Worring 2009].

Hybrid multimedia retrieval
Validity and Qualification
IR applications that allow access to multimedia content (sound, video, still images, ...) Valid for medical images, video retrieval
Dependencies
Text-based multimedia retrieval, content-based multimedia retrieval
Action
Use content-based retrieval to refine results from textual search when possible
Expected Impact
Increases retrieval effectiveness. Combines the advantages of both approaches.
Support of Claim
[Li et al. 2009][Müller 2010] [Snoek & Worring 2009]
Tested Configurations

5.3 End User Interface

The focus on IR applications instead of IR systems in the narrower sense illustrates the important role that the user interface has. The IR application typically supports a knowledge-intensive business process, and the user interface has to support the entire IR cycle as described in the introduction. Even though evaluation following the Cranfield paradigm takes a narrower view of information retrieval, abstracting from users, and concentrating on matching the (coded) query to the (indexed) items, the CLEF evaluation campaign has a strong heritage of considering the user interface as well, through the inclusion of an interactive track ("iCLEF") early on in its existence. Best practices derived from the experiences in iCLEF have been presented in [Braschler & Gonzalo 2009]. We base the following recommendations on that work, but have tried to put the selected recommendations into the extended format of this report, showing more focus on their applicability to different use case domains, specifically the ones considered in Promise.

Information retrieval is, disregarding the possibility that a bored user may be just "poking around", normally a means to an end. The users have information needs, which they try to satisfy by using the IR application. When considering the IR cycle, as given in the introduction of this report, the processing of the results has an important role: it is at this stage, that the user finds new (relevant) information, and potentially gains new insight into his/her information need, allowing a reformulation of the query (or a conclusion of the retrieval task). How the items are presented to the user thus greatly influences his or her ability to quickly carry out a "document selection" task, i.e., to locate those items in the result list that are promising for further inspection [Khan et al. 2009]. What details of the items should be shown in the result list is often very specific to the IR application. The most widespread approaches are either the use of a document snippet (a short excerpt of a textual document, usually giving some context "window" around the most pertinent matches of query terms, i.e., a "query-biased summary" [Tombros & Sanderson 1998]) or the display of associated metadata.

Document snippets
Validity and Qualification
Dependencies
Action
Offer document snippets (query-biased summaries) in the result list
Expected Impact
Faster document selection
Support of Claim
[Khan et al. 2009], [Tombros & Sanderson 1998]
Tested Configurations

Presentation of items needs to be revisited in the multilingual (cross-language) case. If the system returns items in languages other than the one used by the user, there is a large probability that the user cannot readily understand the content of the item. Full manual translation of the item is certainly an option if the user believes in its relevance, but is costly and cannot be obtained for all items prior to document selection. It is thus important to present the user with some surrogate for the item that allows determining relevance accurately and thus narrowing the result set for further inspection. Similar considerations are valid when items are simply not translatable (such as in the "cultural heritage" use case domain), or when very high requirements are connected to the translated form of the item, such as in the "search for innovation" use case domain, where slight translation errors can lead to misinterpretation of essential statements. Work in iCLEF has shown that high-quality, summarized information is best suited to this task, i.e., not full machine translation of the item, even where possible, due to the noisy nature of that approach, but the presentation of noun phrases, relevant passages, and key concepts, will lead to faster, equally accurate document selection by the user [Oard et al. 2004].

Multilingual document summaries
Validity and Qualification
Especially true for use case domains such as cultural heritage and search for innovation, where full-document translation is often not feasible (cultural heritage: untranslatable artifacts, search for innovation: very high requirements with respect to precision of translation)
Dependencies
Action
Offer translated document summaries, containing the most important noun phrases, relevant passages, and key concepts
Expected Impact
Faster document selection, no loss of precision
Support of Claim
[Braschler & Gonzalo 2009], [Oard et al. 2004]
Tested Configurations

In cases where the content of the item itself cannot be presented in summarized form for inspection in the result list, metadata is an appropriate fall-back. Some of the most pertinent metadata fields may either be largely language-independent (some named entities), or mappable across languages (categories etc.) [Braschler & Gonzalo 2009]. The use of document/item summaries and metadata can be combined.

(Language-independent) metadata
Validity and Qualification
High-quality full-text translation is not available or suitable for languages and/or use case domain involved
Dependencies
Document summaries
Action
Use (language-independent) metadata as fall-back from full text translation
Expected Impact
Availability to show users document/item surrogates in result list even if content cannot be summarized
Support of Claim
[Braschler & Gonzalo 2009]
Tested Configurations

When looking at a multilingual or multimedia result list, ranking by relevance is not necessarily the best option in all cases. While users today are used to the ranked list paradigm, a mixed-language or mixed-media result may lead to preferences that may overrule relevance: users may opt to inspect an item with lower relevance ranking first if it is in their preferred language or medium [Braschler & Gonzalo 2009]. It is therefore necessary to offer flexible ways to reorganize the result list: by language, by medium, and – depending on the use case domain – by other criteria such as origin, age etc.

Result list sorting
Validity and Qualification
Cultural heritage: offer at least language and country
Dependencies
Action
Offer multiple ways to sort the result list; adapt to requirements of use case domain
Expected Impact
More effective item selection by the user
Support of Claim
[Braschler & Gonzalo 2009]
Tested Configurations

In multilingual retrieval systems and applications it is, from time to time, nearly unavoidable to show the complexity of bridging between languages to the user. Ideally, languages that users have no active or passive competency in should be hidden; making the system "transparent" to the users as far as language use goes. In practice, this goal is not achievable: formulations in different languages cannot in every case be mapped exactly, due to different ambiguities (homonyms etc.) in each language. However, in cases where mechanisms are available to limit human involvement, such as blind relevance feedback (versus human-assisted relevance feedback) for query refinement, this is preferable. Similarly, query translation should work without user assistance by default, only offering a user-assisted alternative on request [Braschler & Gonzalo 2009] [Petrelli et al. 2003]

Avoid user involvement in query refinement
Validity and Qualification
Multilingual scenarios
Dependencies
Action
Implement self-contained query refinement (e.g., blind relevance feedback) instead of interactive refinement techniques when potential for user confusion exists (e.g., if user does not understand the document language)
Expected Impact
Application more transparent to the user
Support of Claim
[Braschler & Gonzalo 2009]
Tested Configurations

As has been pointed out already in the section on best practices for systems and applications, it is important to carefully integrate the core business entities into the IR application, thus providing better indexed representations of the items to the system. Similar considerations are valid for the user interface side: by linking additional sources and resources, making them available through the interface, users can be assisted in query formulation [Braschler & Gonzalo 2009]. This is especially true in specialized use case domains.

Provide additional context for matching items
Validity and Qualification
Cultural heritage, search for innovation (technical fields), other domain-specific retrieval applications
Dependencies
Action
Link additional sources and resources that provide context information (e.g., encyclopaedic content, maps, information on named entities...)
Expected Impact
Better ability of users to formulate queries in domain-specific contexts, may especially help in cross-language scenarios.
Support of Claim
[Braschler & Gonzalo 2009]
Tested Configurations

5.4 Evaluation

This section discusses best practices concerning learning to rank [Liu 2009]. Learning to rank has become the preferred way of approaching web search, where there is a huge, heterogeneous and redundant collection to be searched. Other aspects than mere relevance to a query play a large role, for example “spamminess” of documents or how many people have linked to a web page. In learning to rank, typically first a set of documents are retrieved by one or more state of the art retrieval algorithms. Then, this set of documents is re-ranked using a function of several features which is learned during a training stage, using queries for which relevance judgments are available. In the learning phase, the function is optimized such that some loss function is minimized. There are three main approaches to do this: (i) pointwise, (ii) pairwise, (iii) listwise. In pointwise learning to rank the loss function calculates the difference in relevance and predicted relevance for each relevant document. In pairwise learning to rank, the loss function calculates if pairs of retrieved documents are ordered correctly in the result list. In listwise learning to rank, an evaluation metric is computed over the entire ranked list, e.g., a standard information retrieval metric such as MAP. In many cases, listwise learning to rank seems to be the most promising approach.

Improving ranked lists
Validity and Qualification
Learning to rank approaches to ad hoc search.
Dependencies
Action
Prefer listwise learning to rank over pairwise and pointwise learning to rank.
Expected Impact
Better performance
Support of Claim
[Liu 2009]
Tested Configurations
OSHUMED corpus with 1 set of topics, GOV corpus with 6 sets of topics used in TREC 2003 and 2004 Web tracks.

Learning to rank is a machine learning problem. An important aspect in any machine learning problem is to design good features; this is often referred to as feature engineering.

Deriving features
Validity and Qualification
Learning to rank
Dependencies
Action
Derive features from several retrieval algorithms.
Expected Impact
Better performance
Support of Claim
[Liu et al. 2009]
Tested Configurations

Example benefits of putting well known retrieval algorithms to good use are:

- increasing recall base
- possibility of derived features such as the number of retrieval algorithms that retrieved a document.
- In structured retrieval, the possibility to calculate different retrieval algorithm scores for different fields: a learning to rank algorithm can then give importance to different retrieval algorithms on different fields.

A large benefit of learning to rank is that in addition to retrieval algorithm scores for the query, other features of documents can be seamlessly integrated. For example, in web search, the PageRank [Brin & Page 1998] score of a web page is by now ubiquitous as a feature in learning to rank. Other features could include the likelihood of a document to be spam, or credibility of authors [Weerkamp et al. 2012]

Feature selection
Validity and Qualification
Learning to rank approaches to ad hoc search.
Dependencies
Action
Use features that are quality indicators of documents, e.g., probability of being spam (spamminess), PageRank, freshness and so on. Use any metadata associated with documents, e.g., if there is authorship information, additional features may be derived such as reputation of author, credibility, and so on.
Expected Impact
Better performance
Support of Claim
[Liu et al. 2009]
Tested Configurations

Once the set of features is determined, it should be plotted. If a training set with document relevance information is available, it can be a good idea to plot overlapping histograms for each feature: one histogram for non-relevant documents, and one histogram for relevant features. In this way, it can be immediately visualized if a feature on its own shows promise for distinguishing between relevant and non-relevant documents. Also, it helps to check if there are any outliers in the feature values, these could for example be caused by errors in the feature calculation. Before feeding the feature values to a machine learning algorithm, in most cases it is necessary to normalize features values. There are many different ways to normalize features, but most machine learning algorithms require some form of normalization to function reliably.

Feature normalization
Validity and Qualification
Learning to rank approaches to ad hoc search, machine learning applications in general.
Dependencies
Action
Normalize features before feeding values into machine learning algorithms.
Expected Impact
More reliable performance, better performance.
Support of Claim
An excellent resource on questions on normalization, with backlinks to discussion on the more general issue of measurement theory is ftp://ftp.sas.com/pub/neural/FAQ2.html#A_std
Tested Configurations
SVM based algorithms, Neural networks, and so on.

6 Verification through Stakeholder Interviews

The stakeholder interviews conducted in conjunction with Promise task 2.3 ("Validation of use cases") and task 2.6 ("Technology take-up group") can illustrate some of the limitations in applying the best practices to operational settings. It should be kept in mind that the systems underlying the search functionalities of the information retrieval applications are built on the foundation of retrieval models that have a number of assumptions that may be invalidated in a concrete setting. Some of the more prominent assumptions include:

- the user is looking for "relevant" documents
- the user is looking for "unknown" information (starting from an information need, the user tries to solve a "problem" which s/he has no solution for yet)
- the number of relevant items is unknown

Digesting the output from the stakeholder interviews, we can learn a number of additional "viewing angles" with respect to the use of information retrieval applications. Questions were intentionally broad ranging, with the possibility of "open answers" by the participants". The questionnaire addressed -among other things- the following points:

- Data: the kind of data covered by the document collection, the indexing process
- Systems: the features of the retrieval applications
- Users: the different user groups and their main differences
- Sessions: the characteristics of typical search sessions (patterns, lengths, etc.)
- Evaluation: efforts for evaluating and monitoring system performance

As a result, we have identified the following issues that so far are not necessarily directly reflected in the best practices as presented in the deliverable. It will be a part of future extensions to the best practice framework to incorporate more use case domains that will better and additionally cover some of the following limitations:

- Result granularity may vary widely from one use case domain to the next. This is, e.g., the case when streaming media, such as video transcripts, are searched. It may not be possible with present technology to locate relevant content directly, and instead necessary to implement a hybrid search/browse approach, where the user is pointed to the right "vicinity" of the content, and then has to browse the stream onwards to locate the actual content (identified e.g., in interview with the National Archive of Sound and Vision, Sweden)
- Operators of information retrieval applications may be forced by legislation or other mandatory concerns to monitor the usage of the system by the user (detection of "inappropriate usage"). This issue is poorly covered by academic research so far (identified, e.g., in interview with the National Archive of Sound and Vision, Sweden)
- Similarly, issues of auditability, legal regulations, or copyright concerns may dictate the use (or non-use) of certain features and/or algorithms. This issue is also poorly covered by academic research today (identified, e.g., in interview with the National Archive of Sound and Vision, Sweden).
- "Query-based" searching with the goal of the maximisation of "relevant" content (see some of the assumptions given above) may not actually be congruent with the real goals of the user population. E.g., users may be much more interested in the recency or social impact of content than in its topical relevance. There is increasing research into moving beyond topical relevance, and onwards to other retrieval criteria. Some of this research is addressed in some of the best practice recommendations, such as "retrieval paradigm" and "result list sorting" (identified, e.g., in interview with the Finnish commercial TV station MTV3)
- The data and information contained in the collection made accessible through the application may be specialized to such an extent, that it is not possible for the casual user (even though being a domain expert) to code their information needs suitably for the system nor to interpret the system output without aid (identified, e.g., in interview with the Swedish Institute of Communicable Disease Control)
- The result list may be the only thing that is actually consumed by the users, i.e., the users never really "click through" to the underlying document (an interesting consequence of this phenomenon is that it is very hard to derive meaningful analysis from log files of the application). This is especially true when the result list presentation is visually rich, such as in picture search, where it may contain thumbnails that very well present the entirety of the content of the underlying "document". This is somewhat in contradiction with the best practices for result list presentation, which assume that the underlying document is the ultimate goal of the user (identified, e.g., in interviews with the Web image search provider picsearch.com).

7 Conclusions

This report presents the result of Promise task 2.5 "Best Practices in Multilingual and Multimedia Information Access". We present a list of best practice recommendations spanning aspects from the IR system and IR application to the user interface and to evaluation. The recommendations have been elaborated to generalize as broadly as possible. There are limitations as to how widely any recommendation is applicable, however, especially with today's use of information retrieval technology in applications ranging from Web search services to recommender systems, or library search systems to topic detection applications. We have tried to address this issue by using a new structure to present the best practice recommendations, which includes fields on qualifications, depending on the use case domains of an IR application, and which will accommodate in the future information pointing directly to related experiments.

The aim to cover information retrieval technology so broadly leads to a practical impossibility to consider all possible use case domains and all possible aspects, at least initially. The development of new extensions to the DIRECT framework in Promise will be helpful to enrich the best practices with new information, such as pointers to experiments, or to discover new candidates for additional best practice recommendations. Still open is also the question on how implement a "life cycle" for the recommendations. Can we anticipate which recommendations will be long-lived or which will become obsolete over the long term?

As the preceding section on validation through stakeholder interviews discusses, new use case domains bring entire new focus areas that have not been fully covered by the set of recommendations given in the report. Specifically, new, emerging forms of IR applications constantly challenge the (implicit) assumptions we have about the workings of IR systems. One example is the move from a single user to groups of users that interact (possibly outside the application context proper) – we hope that publication and dissemination of the present report will lead to additional new suggestions for extending the work in such directions.

Acknowledgements

We would like to thank Julio Gonzalo from UNED for his permission to start with his work in [Braschler & Gonzalo 2009] for the user interface section. Many thanks go all Promise partners that contributed directly and indirectly to the report.

References

[Amati & Rijsbergen 2002] Amati G, Rijsbergen CJV (2002) Probabilistic models of information retrieval based on measuring the divergence from randomness. ACM Trans. Inf. Syst. 20(4): 357–389

[Azzopardi et al. 2010] L. Azzopardi, W. Vanderbauwhede, and H. Joho, Search system requirements of patent analysts, Proceeding of the 33rd international ACM SIGIR conference on Research and development in information retrieval, ACM, 2010, pp. 775–776.

[Barcala et al. 2002] Fco. Mario Barcala, Jesús Vilares, Miguel A. Alonso, Jorge Grãna, Manuel Vilares, Tokenization and Proper Noun Recognition for Information Retrieval. In: Database and Expert Systems Applications, 2002. Proceedings. 13th International Workshop on IEEE, 2002.

[Becks et al. 2009] Daniela Becks, Christa Womser-Hacker, Thomas Mandl und Ralph Kölle: Patent Retrieval Experiments in the Context of the CLEF IP Track 2009, Multilingual Information Access Evaluation, Lecture Notes in Computer Science, 2010, Volume 6241/2010, 491-496.

[Brin & Page 1998] Brin, S. and Page, L. The anatomy of a large-scale hypertextual Web search engine. Computer networks and ISDN systems, 1998

[Braschler 2004] Braschler M, Combination approaches for multilingual text retrieval. Information Retrieval, 7(1-2):183–204, 2004.

[Braschler & Ripplinger 2004] Braschler M, Ripplinger B, How effective is stemming and decompounding for German text retrieval? Inf. Retr. 7(3–4): 291–316, 2004

[Braschler & Gonzalo 2009] Braschler M, Gonzalo J, Best practices in system and user-oriented multilingual information access. 2009

[Broder 2000] Broder AZ, Identifying and filtering near-duplicate documents. In: Proc. 11th Annual Symposium on Combinatorial Pattern Matching (COM '00). Springer-Verlag, London: 1–10, 2000

[Clarke et al. 2008] C.L.A. Clarke, M. Kolla, G.V. Cormack, O. Vechtomova, A. Ashkan, S. B¨uttcher, and I. MacKinnon, Novelty and diversity in information retrieval evaluation, Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval, ACM, 2008, pp. 659–666.

[Cleverdon 1967] Cleverdon CW (1967) The Cranfield tests on index language devices. In: Aslib Proc. 19(6): 173–192

[Dolamic et al. 2008] Dolamic, L., Fautsch, C., Savoy, J.: UniNE at CLEF 2008: TEL, Persian and Robust IR. In Peters, C. (Ed).: Working Notes for the CLEF 2008 Workshop.

[Dolamic & Savoy 2009] Dolamic L, Savoy J When stopword lists make the difference. J. of Am. Soc. for Inf. Sci. 61 (1): 200–203, 2009

[El-Khair 2007] El-Khair IA, Arabic information retrieval. Annu. Rev. of Inf. Sci. and Technol. 41(1): 505–533, 2007

[Harman 1991] Harman D, How effective is suffixing?. J. of the Am. Soc. for Inf. Sci. 42(1): 7–15, 1991

[Hull 1996] Hull DA, Stemming algorithms - A case study for detailed evaluation. J. of the

Am. Soc. For Inf. Sci. 47(1): 70–84, 1996

[Kankanhalli &Rui 2007] Mohan S. Kankanhalli and Yong Rui, Application potential of multimedia information retrieval, 2007.

[Khan et al. 2009] Khan R., Mease D., and Patel R.. The impact of result abstracts on task completion time. In Workshop on Web Search Result Summarization and Presentation, WWW'09.

[Kim et al. 2011] Y. Kim, J. Seo, and W.B. Croft, Automatic boolean query suggestion for professional search, Proceedings of the 34th international ACM SIGIR conference on Research and development in Information, ACM, 2011, pp. 825–834.

[Kleinberg 1999] Kleinberg J., Authoritative sources in a hyperlinked environment, Journal of the ACM, 46(5),1999, 604-632.

[Lazarinis et al. 2008] F. Lazarinis and E.N. Efthimiadis, Measuring search engine quality in image queries in 10 non-english languages: an exploratory study, Proceeding of the 2nd ACM workshop on Improving non english web searching, ACM, 2008, pp. 89–92.

[Li et al. 2009] Qing Li, Yi Zhuang, Jun Yang, and Yueting Zhuang, Multimedia information retrieval at a crossroad, Encyclopedia of Multimedia Technology and Networking, Second Edition, IGI Global, 2009, pp. 986–994.

[Liu 2009] Liu, T. Y.: Learning to Rank for Information Retrieval, Foundations and Trends in Information Retrieval, Vol. 3, Number 3, pp. 225-331, 2009. Now Publishers.

[Mandl et al. 2008] Mandl T, Womser-Hacker C, Di Nunzio GM, Ferro N (2008) How robust are multilingual information retrieval systems? Proc. SAC 2008 - ACM Symposium on Applied Computing: 1132–1136

[McNamee & Mayfield 2003] McNamee P and Mayfield J, JHU/APL experiments in tokenization and non-word translation. In: Proc. 4th Workshop of the Cross-Language Evaluation Forum (CLEF 2003). Springer- Verlag LNCS 3237: 85–97, 2003

[McNamee & Mayfield 2004] P. McNamee and J.Mayfield, Character N-Gram Tokenization for European Language Text Retrieval. Inf. Retr. 2004

[Moulinier & Williams 2005] I. Moulinier and K. Williams, Report on thomson legal and regulatory experiments at CLEF-2004, In: Multilingual Information Access for Text, Speech and Images, 920—920, Springer, 2005

[Müller 2010] Henning Müller, ImageCLEF experimental evaluation in visual information retrieval, Springer, Heidelberg, 2010.

[Oard et al. 2004] Oard, D., Gonzalo, J., Sanderson, M., Lopez-Ostenero, F. and Wang, J. (2004)

Interactive cross-language document selection. Information Retrieval, 7 (1-2)pp. 205-228.

[Perez et al. 2009] Joaquın Perez-Iglesias, Guillermo Garrido, Alvaro Rodrigo, Lourdes Araujo, and Anselmo Penas, Information retrieval baselines for the respubliqa task, Proceedings of the 10th cross-language evaluation forum conference on Multilingual information access evaluation: text retrieval experiments (Berlin, Heidelberg), CLEF'09, Springer-Verlag, 2009, pp. 253–256.

[Peters et al. 2012] Carol Peters, Martin Braschler, Paul Clough: Multilingual Information Retrieval: From Research To Practice, Springer, 2012, ISBN 3642230075

[Petrelli et al. 2003] Petrelli D, Demetriou G, Herring P, Beaulieu M, Sanderson M (2003) Exploring the effect of query translation when searching cross-language, In Peters C et al. (Eds.) Advances in Cross-Language Information Retrieval. Springer, LNCS 2785.

[Pfeifer et al. 1996] U. Pfeifer, T. Poersch and N. Fuhr, Retrieval effectiveness of proper name search methods. In: Information Processing & Management, 1996

[Ripplinger 2000] B. Ripplinger, The Use of NLP Techniques in CLIR. In Peters, C. (Ed): First Results of the CLEF 2000 Cross-Language Text Retrieval System Evaluation Campaign. Working Notes for the CLEF 2000 Workshop, ERCIM-00-W01.

[Robertson 1977] Robertson SE (1977) The probability ranking principle in IR. J. of Doc. 33(4): 294–304

[Robertson et al. 1980] Robertson SE, van Rijsbergen CJ, Porter MF (1980): Probabilistic models of indexing and searching. Proc. ACM SIGIR conference on research and development in information retrieval (SIGIR '80). ACM, New York: 35–56

[Runeson et al. 2007] P. Runeson, M. Alexandersson, and O. Nyholm, Detection of duplicate defect reports using natural language processing, Software Engineering, 2007. ICSE 2007. 29th International Conference on, Ieee, 2007, pp. 499–510.

[Rüger 2009] Stefan Rüger, Multimedia resource discovery, Information Retrieval: Searching in the 21st Century (Ayse Goker and John Davies, eds.), JohnWiley and Sons, Chichester, 2009, pp. 39–62.

[Salton et al. 1975] Salton G, Wong A, Yang C, A vector space model for automatic indexing. Commun. of the ACM 18(11): 613–620, 1975

[Silverstein et al. 1999] Craig Silverstein, Hannes Marais, Monika Henzinger, and Michael Moricz. 1999. Analysis of a very large web search engine query log. SIGIR Forum 33, 1 (September 1999), 6-12.

[Savoy & Dolamic 2009] Savoy J, Dolamic L (2009) How effective is Google's translation service in search? Commun. of the ACM 52(10): 139–143

[Singhal et al. 1996] Singhal A, Buckley C, Mitra M (1996) Pivoted document length normalization. In: Proc. ACM SIGIR conference on research and development in information retrieval (SIGIR '96). ACM, New York: 21–29

[Snoek & Worring 2009] C. G. M. Snoek and M. Worring, Concept-based video retrieval, Foundations and Trends in Information Retrieval 4(2009), no. 2, 215–322.

[Tombros & Sanderson 1998] Tombros A. and Sanderson M.. Advantages of query biased summaries in information retrieval. In SIGIR, pp. 2–10, 1998.

[Voorhees 2002] Voorhees EM (2002) The philosophy of information retrieval evaluation. In: Evaluation of cross-language information retrieval systems: 2nd workshop of the Cross-Language Evaluation Forum, CLEF 2001, Springer, LNCS 2406: 355–370

[Walker et al. 1998] Walker S, Robertson SE, Boughanem M, Jones GJF, Spärck Jones K (1998) Okapi at TREC-6, automatic ad hoc, VLC, routing, filtering and QSDR. In Voorhees EM, Harman DK, (eds.) The Sixth Text REtrieval Conference (TREC-6). NIST Special Publication 500–240: 125–136

[Weerkamp et al. 2012] Weerkamp W., de Rijke M., "Credibility-inspired Ranking for Blog Post Retrieval", Information Retrieval Journal, vol. 15, no. 3--4, pp. 243-277, June, 2012

[Zipf 1949] Zipf, G.K. . Humun Behavior and the Principle of Least Effort. Cambridge, MA: Addison-Wesley., 1949