Webspace retrieval performance experiment

Roelof van Zwol Peter M.G. Apers

University of Twente,
Centre for Telematics and Information Technology,
Department of Computer Science
P.O.box 217, 7500 AE, the Netherlands
{zwol, apers}@cs.utwente.nl

Abstract

Finding relevant information using search engines that index large portions of the World-Wide Web is often a frustrating task. Due to the diversity of the information available, those search engines will have to rely on techniques, developed in the field of information retrieval (IR).

When focusing on more limited domains of the Internet, large collections of documents can be found, having a highly structured and multimedia character. Furthermore, it can be assumed that the content is more related. This allows more precise and advanced query formulation techniques to be used for the Web, as commonly used within a database environment. The Webspace Method focuses on such document collections, and offers an approach for modelling and searching large collections of documents, based on a conceptual schema.

The main focus in this article is the evaluation of a retrieval performance experiment, carried out to examine the advances of the webspace search engine, compared to a standard search engine using a widely accepted IR model. A mayor improvement in retrieval performance, measured in terms of recall and precision, up to a factor two, can be achieved when searching document collections, using the Webspace Method.

1 Introduction

Over time the Internet has grown into an ever more tangled resource of information. The state-of-the-art means for finding information are text-based search engines like Alta-Vista and Google, hierarchical indexes and directories like Yahoo!. These search engines have to base their tools solely on information retrieval techniques, due to the unstructured nature of the Internet. The diversity, irregularity, and incompleteness of the data involved, make it impossible to use database technology at this global level. Besides that the data has the tendency to change rapidly over time.

However, when focusing on smaller domains of the WWW, database techniques can be invoked successfully to enhance the precision of the search process. On such domains, large collections of documents can be found, containing related information. Although the conditions are better, one still has to deal with the semi-structured and multimedia character of the data involved. The *Webspace Method* [vZA99] focuses on such domains, like Intranets, digital libraries, and large web-sites.

The goal of the *Webspace Method* is to provide sophisticated search facilities for web-based document collections, using existing database techniques. To achieve this, three stages are identified for the Webspace Method. The first stage deals with conceptual modelling of web-data. During the second stage, both conceptual and multimedia meta-data is extracted from the document collection. Finally, in the last stage, the Webspace Method introduces a new approach to query a collection of web-based documents.

The key behind the Webspace Method is the introduction of a *semantical level*, which provides a conceptual description of the content of a webspace. This semantical level consist of a collection of concepts,

which are defined in an *object-oriented* schema. The object-oriented schema is also referred to as the *webspace schema*. The second level of the Webspace Method is called the *document level*. This level is formed by the web-based document collection. Each document on the document level of a webspace consists of a view, corresponding to (a part of) the webspace schema, to assure the relation between the semantical and physical levels of the Webspace Method. The webspace schema defined during the first stage, is also used to extract the relevant meta-data during the second stage, and to formulate queries over a *webspace*. A webspace is formed by both the semantical and the document level.

Revolutionary within the scope of search engines and query formulation over document collections is that the Webspace Method allows a user to integrate (combine) information stored in several documents in a single query. At this point traditional search engines, e.g. Alta-Vista and Google, are only capable to query a single document at a time. As result, the query will return the requested conceptual information as a view on the webspace schema, rather than returning a collection of relevant document URLs.

The main focus of this article forms the retrieval performance experiment. To evaluate the retrieval performance of the webspace search engine, and in particular the contribution of conceptual modelling to the retrieval process, a retrieval performance experiment is carried out. It measures the increase in performance of the webspace search engine, compared to the traditional way of querying a collection of documents, using a standard search engine. For this purpose, the standard search engine is equipped with the same IR model for text retrieval as the webspace search engine. A third search engine is also evaluated, which implements a fragmented variant of the webspace search engine. The motivation for this fragmented variant, along with a discussion of the implementation is given in Section 5.

The evaluation method used for the experiment is related to the *TREC test collection*, as described in Section 4.2. The second part of this article describes the experimental setup, which provides insight in the test collection, setup for this experiment. It is based on the 'Lonely Planet' web-site. The original web-site can be found at [Pub01], and contains 6500 documents with structured information about destinations all over the world, divided in regions, countries, and cities. Furthermore, it contains a large collection of postcards, send in by travellers, containing the experiences of those travellers, while staying at a certain destination.

The results of the experiment show a high increase in retrieval performance, measured in terms of *recall* and *precision*. On average, both webspace search engines perform up to a factor two better, than the standard search engine. Furthermore it proves that the search engines based on the Webspace Method are capable to find information that can not be found by (standard) search engine, due to the conceptual model introduced by the Webspace Method.

Related work

Modelling data on the web is an ongoing research area, where many research projects have been positioned around. Closely related to our approach is the Araneus project [MMA99] where also existing database technology is applied to the WWW. This project is also concerned with handling both structured and semi-structured documents. The main difference with our approach is that we aim at combining concept-based search with content-based information retrieval, to come up with more precise query formulation techniques for data on the web. Others [AO98, CCD+99], like in XQuery [CFR+01] use the structure of an XML document as their model. This allows them to search for patterns and structure in the XML data, but does not allow them to formulate content based (IR) queries. In [FG00, HTK00], about XIRQL and searching text-rich XML documents with relevance ranking, the focus is on XML and information retrieval. But these approaches do not use a conceptual model, and integrate the IR model only partially. Of course in the field of information retrieval, and multimedia databases many sophisticated models are proposed. We do not aim to come with better IR techniques, but aim to combine existing IR techniques with conceptual modelling, using a database-oriented approach. For those interested in information retrieval and MM-DBMS, we refer to [BYRN99, DF98, dVW99], where these matters are discussed.

Organisation

In the remainder of this article, the ideas behind the Webspace Method are explained in Section 2. Backgrounds on text retrieval are explained in Section 3. Before discussing the experiment a short discussion of retrieval performance evaluation is presented in Section 4 to explain some of the backgrounds of the experimental setup, presented in Section 5. The results of the retrieval performance experiment are discussed in Section 6. Finally, we will come to the conclusions in Section 7.

2 Ideas behind the Webspace Method

In the introduction we already argued that database techniques cannot be applied straightforwardly to search for data on the Web, since the Web is not a database [MMM97]. Therefore the Webspace Method aims at smaller domains of the Internet, where large collections of documents can be found containing related information. Such data can typically be found on large Intranets, web-sites and digital libraries. The Webspace Method for modelling and querying web-based document collections is developed for such collections. It combines conceptual search with content-based information retrieval to obtain more precise and advance query formulation techniques for web-data.

The Webspace Method consists of three stages: a modelling stage, a data-extraction stage, and a query stage. The modelling stage is used to define a webspace at both the semantical and document level. Next, during the indexing stage, meta-data is extracted from a webspace, and stored into the object server used by the webspace search engine. Finally, during the query stage, a webspace can be queried using the conceptual schema (webspace schema), which allows a user to formulate his information need in terms of concepts defined in the webspace schema.

2.1 Modelling a webspace

For each webspace, it is possible to identify a finite set of concepts, which adequately describe the content of a webspace at a semantical level [vZA00a]. During the modelling stage of the Webspace Method, these concepts are modelled in an object-oriented schema, called the webspace schema. At the document level the content is stored in XML documents, or in a content DBMS. In either case the representation of the document, as presented to the end user, forms a materialised view. To create such a view, it requires that the author specifies a structure of the document, provides the document's content, as well as the default presentation.

2.1.1 Webspace schema

Each concept defined for a webspace is referred to by a unique name. Furthermore, each concept should be defined in the webspace schema. Going towards the test collection used for the retrieval performance experiment, the webspace schema presented in Figure 1 is setup to semantically describe the content of the 'Lonely Planet' webspace.

It contains ten class definitions with their attributes and the associations between the classes. The underlined attributes form the uniques keys of a web-class. The attributes having their type displayed are of a multimedia class, which realises the integration of multimedia objects into the conceptual framework The dotted boxes, illustrate how the information contained in the XML documents at the document level of a webspace are spread over the different conceptual classes.

2.1.2 Materialised Views

Each document found on the document level forms a *materialised view* on the webspace schema. Thus, the dotted box containing the classes *Postcard* and *Destination* (Figure 1) forms a view on the webspace schema, since it describes only a part of the schema. All documents that contain information, related to

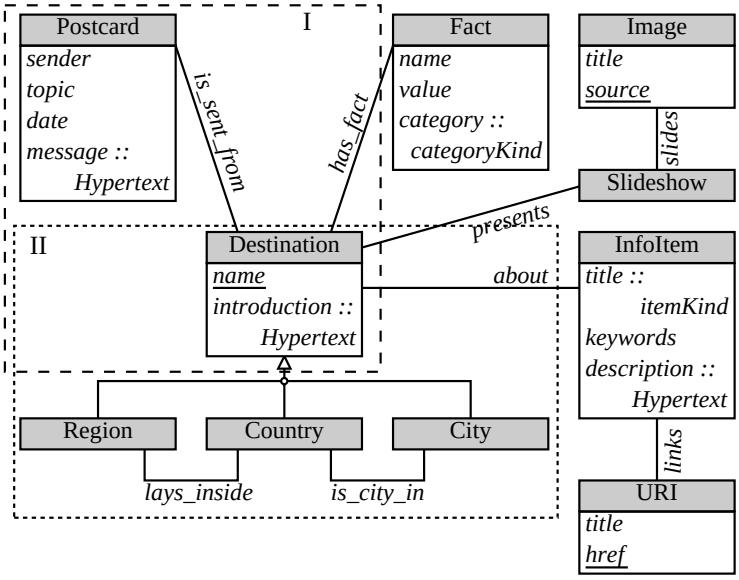

Figure 1: Webspace schema for the 'Lonely Planet' case-study.

(exactly) this part of the schema are materialising this view. Other documents might contain different materialised views, while the collection of materialised views is likely to have some overlap, as is the case for boxes I and II of Figure 1.

To derive a materialised view from the webspace schema, i.e. an XML-document, several steps will have to be taken in order to define the structure of the document explicitly. This procedure is described in more detail in [vZA00a] and implemented by the *webspace modelling tool*.

2.2 Indexing a webspace

During the extraction stage of the Webspace Method the object server needed querying a webspace is set-up. Depending on the webspace schema and the types of multimedia used for a webspace, daemons need to be administered in the Daemon Data Dictionary. A *daemon* is a program performing a single task on a web object. In general, when the object is a multimedia object, the daemon will try to extract relevant meta-data and store it in the object server. But also for the storage of conceptual objects daemons are used. A detailed description of the DDD and its role in the extraction stage is given in [vZA00b].

2.3 Querying a webspace

The main contribution of the Webspace Method is that it provides a new approach for querying web-based document collections. The basis for this is formed by the webspace schema, which allows powerful query formulation techniques, as known within a database environment, to be invoked on the content of a webspace. The query formulation techniques, implemented by the webspace search engine, allow queries to be formulated, by using the webspace schema, over the content stored in one or more documents. Secondly, it allows users to directly extract the relevant parts of one or more documents as the result of a query, instead of a document's URL.

The webspace schema developed during the modelling stage of the Webspace Method is provided to the user, to specify his information need. To allow a user to formulate these relatively complex queries over a webspace a graphical user interface (GUI) is used, which guides the user through the query formulation process. The GUI provides a visualisation of the classes contained in the webspace schema, which is used to determine the query skeleton. Next, the constraints are formulated, and the user specifies the information that is returned as the result of the query.

Since the user specifies his information need by selecting concepts from the webspace schema, it can formulate queries that combine information originally stored in several documents, i.e. materialised views. Using the overlap between the materialised views, allows the webspace search engine to find information that can not be found by search engines that do not rely on a conceptual schema. Crucial for the success of the Webspace Method is the schema-based approach, but also the integration with the information retrieval, which allows us to benefit from the existing knowledge in that area.

3 Text retrieval

A substantial part of the information retrieval is devoted to text retrieval. In this section the focus is on those IR systems. Section 3.1 describes how documents are preprocessed to derive a set of terms, that form the input for many of the IR models. One of these models, the vector space model, is described in more detail in Section 3.2. Section 3.3 discusses how this IR model can be translated to relational tables, which allow a database-oriented approach for text retrieval as proposed in Mirror [dV99], a content and multimedia DBMS.

3.1 Logical view of a document

Most IR models base their retrieval strategy on a set of *index terms* which are defined in the logical view of a document. A set of index terms is best described as a collection of representative keywords. A preprocessing phase is often used, to derive the logical view of a document from the original full text. In the preprocessing phase a sequence of *text operations* are usually invoked to generate the set of index terms. In Figure 2 the steps of the preprocessing phase are depicted, as presented in [BYRN99].

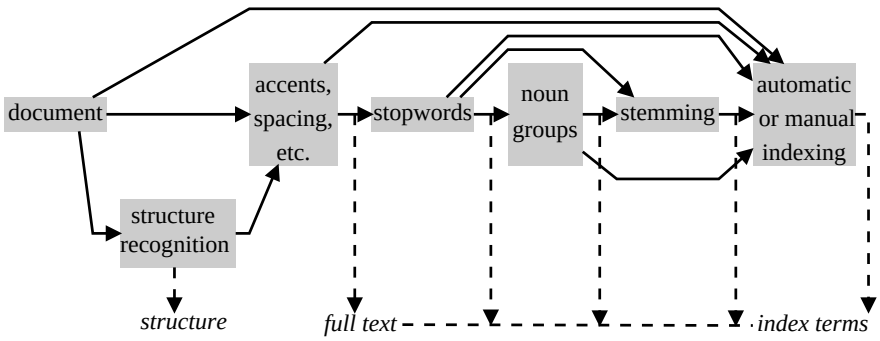

Figure 2: Derivation of the logical view of a document [BYRN99].

Starting on the left-hand side, recognition of the document structure is sometimes used to enhance query formulation in the information retrieval. However, most IR systems will start with removing accents, spacing, and other lexical elements. During the following steps, stopwords are eliminated from the text, noun groups are detected, and stemming of terms is applied, before the term indexing selection process is started. The bottom line of the figure shows how the document's full text is gradually transformed into the index terms used as input for the retrieval strategy.

3.1.1 Text operations

Many text-operations carried out during the document preprocessing phase have been proposed which often aim at reducing the size of the collection of index terms. Two text operations, often used in combination with the vector space model, are discussed here. The first operation is called *eliminating stop words*. This technique reduces the index, by removing irrelevant articles, prepositions, and conjunctions. The second

text operation is *stemming*. Stemming converts index terms to their grammatical root. This technique increases the recall and also reduces the term index.

Eliminating stopwords

Although not a difficult task, the effect of eliminating stopwords is clearly visible. If a word occurs in nearly every document, that word is useless when used as a query term. These words are referred to as stopwords. The list of stopwords usually contains articles, prepositions, and conjunctions. The main advantage is that it reduces the term index drastically (up to 40%). Elimination of stopwords can also reduce the recall. For instance the query: '*to be or not to be*' might perform really bad [BYRN99].

Stemming

Stemming is a technique used to convert words to their grammatical root. When analysing a text, each word can occur in many different forms, i.e. plurals, past tense, and more. This makes it hard to match a query term with the document terms. A stem is that portion of a word, which is left after removing the affixes, i.e. prefixes and suffixes. Stemming also reduces the size of the term index, and has a positive influence on the recall. However it sometimes has a negative influence on the precision, since two semantically different words can be reduced to the same stem. A well-known example of a stemming algorithm is the Porter algorithm [Por80, BYRN99, KP96].

Besides eliminating stopwords and stemming, other preprocessing tasks are also frequently used, like selection of terms, and the use of thesauri [BYRN99]. Other techniques which are used in combination with document preprocessing, include document clustering and text compression.

3.2 Vector space model

The *vector space model* [SYW75] ranks documents, by constructing an n-dimensional vector, where n equals the number of index terms in the collection. To compute the relevance of a document to a query, two parameters are used by the vector space model: the *term frequency*, and the *inverse document frequency*.

- **Term frequency**. For each index term and document pair, the term frequency tf is the number of times the term occurs in the document.
- **Inverse document frequency**. For each term, the inverse document frequency idf is the inverse number of documents, in which the term occurs.

To calculate the similarity coefficient SC between a query Q, and document D_i the following definitions are used:

n = the number of distinct index terms in the collection.
N = the number of documents in the collection.
tf_{ij} = the number of occurrences of term t_j in document D_i.
df_j = the number of documents, containing term t_j.
$idf_j = log(N/df_j)$.

The document vector is constructed as a n-dimensional vector, with an entry for each term of the collection. Based on the term frequency, and the inverse document frequency a weight $tfidf$ is assigned to term t_j for a given document D_i:

$$tfidf_{ij} = tf_{ij} \times idf_j$$

The relevance of a query $Q(w_{q1}, w_{q2}, \ldots, w_{qt})$ to document D_i is calculated by the similarity coefficient as:

$SC(Q, D_i) = \sum_{j=1}^{t} w_{qj} \times tfidf_{ij}$ (simple IR systems will use $w_{qj} = 1$)

To compute a complete ranking, the similarity coefficient is calculated for all the documents in the collection.

3.3 Towards a database approach

The vector space model described in the previous section, is used for content-based text retrieval in Mirror, a content-based multimedia DBMS. Its implementation is based on Moa. In the second part of this section it is shown how the vector space model of the previous section is translated into database relations by the Mirror framework. The mayor benefit of Mirror, by using Moa at the logical level, is that the set-oriented nature of a DBMS is preserved. This formed one of the main reasons for existing IR systems to neglect existing database technology. Another issue is that of decrease in performance, when scaling up the document collection. In [BHC+01, BdVBA01] the scalability issues of content-based DBMSs are studied. Using top-N optimisation techniques, including horizontal fragmentation and parallel execution, the response time of content-based multimedia DBMSs can be reduced. These extensions make Moa a good platform for content-based multimedia retrieval, following a database-oriented approach.

3.3.1 About Mirror

The main goal of Mirror [dV99], is to define a platform for *content-independent* multimedia DBMSs. Four requirements have been defined for the Mirror platform which have to be held, in addition to requirements normally defined for DBMSs.

- The content of a multimedia DBMS should be able to describe multimedia objects.
- Querying a multimedia DBMS is an interactive process, based on the principles of *relevance feedback*.
- Query processing should be able to use different representations (features) of the same multimedia object to full-fill the users information need.
- Multimedia query formulation provides content independence.

To achieve a complete integration of content management and databases, the inference network model of information retrieval is adopted and implemented in Moa. The vector space model described in the previous section, is used for content-based text retrieval in Mirror.

The specific domain knowledge, concerning the IR model is implemented at the logical level, using Moa's structural extensibility mechanism. Lower level information retrieval techniques are implemented by extending the Monet database kernel. In addition to text retrieval, the Mirror framework is also tested in some small-scale experiments in the domains of music, and image retrieval, which proves the content independence of the Mirror framework.

Finally, the Mirror framework provides support for distribution of data and operations, and the extensibility of data-types and operations. From a digital library perspective such characteristics are desired in the context of multimedia management [dVED98]. The content providers are usually not the same as the content access providers, or the end-users of a multimedia digital library.

3.3.2 Vector space model in database relations

In Mirror the term frequency and inverse document frequency are described by the database relations:

TF(term,document, tf)

IDF(term,idf)

The terms used in a query are described by the relation Q(term). Query processing is then handed to Moa, where specialised optimisers can be used to deal with the query efficiently.

The set-oriented nature of Mirror's IR query processing is illustrated in the steps below:

1. Initialise the query process, given a query Q.
2. Limit TF and IDF, by matching them with the query terms of query Q (semi-join):
   $\mathsf{TF_Q} = \mathsf{TF} \ltimes \mathsf{Q}$;
   $\mathsf{IDF_Q} = \mathsf{IDF} \ltimes \mathsf{Q}$;
3. Place $\mathsf{IDF_Q}$ values, next to the $\mathsf{TF_Q}$ entries (join):
   $\mathsf{TFIDF_{lineup}} = \mathsf{TF_Q} \bowtie \mathsf{IDF_Q}$;
4. Aggregate the tf and idf into terms of tf∗idf for each term-document pair:
   TFIDF =
   SELECT term, document, tf ∗ idf as tfidf
   FROM $\mathsf{TFIDF_{lineup}}$;
5. Compute the documents ranking, by aggregation all the terms for each document:
   RANK =
   SELECT sum(tfidf) AS ranking, document
   FROM TFIDF
   GROUP BY document
   ORDER BY ranking DESC;

4 Retrieval performance evaluation

Measuring the response time and usage of space of a system are normally the objectives, when carrying out a performance evaluation. In the information retrieval, other metrics, besides time and space are of interest. A user's information need is formulated using a set of terms, and the result of a query contains inexact answers, where, based on the assumed relevance of a document, a ranking is produced. Therefore the main objective of a *retrieval performance evaluation* in the information retrieval is to measure the precision of the answer set given a specific query.

4.1 Retrieval performance measures

In the information retrieval a retrieval performance experiment is based on a *test reference collection* and one or more *evaluation measures*. The test reference collection consists of a collection of documents, a set of information requests (queries), and a set of relevant documents for each information request, provided by specialists.

The performance of a retrieval strategy, implementing a specific IR model, is measured by calculating the similarity between the set of documents found by the retrieval strategy, for each information request, and the set of relevant documents of the test reference collection.

The measures *recall* and *precision* are most frequently used, to measure the retrieval performance of an experiment, evaluating an IR system in batch mode. Figure 3 shows how these measures can be calculated for a specific information request.

Given an information request I, taken from the test reference collection. The set of relevant documents R for that specific information request, then $\mid R \mid$ refers to the number documents in this set. Let A be the answer set of documents found by the retrieval strategy for information request I. Then $\mid A \mid$ is the number of documents in the answer set. Further, let $\mid R_a \mid$ be the number of documents in the intersection of the sets R and A. The measures for recall and precision are then calculated as:

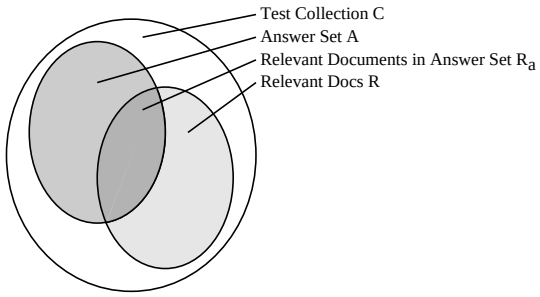

Figure 3: Information request.

$$Recall = \frac{|R_a|}{|R|}$$

$$Precision = \frac{|R_a|}{|A|}$$

The 'perfect' IR system, would have a recall and precision of both 1. Unfortunately such systems do not exist (yet). Furthermore, basing the retrieval performance solely on these two measures would neglect the fact that the answer set of a specific information request is ranked. Retrieval strategies that rank the relevant documents higher should be rewarded for that. This is normally expressed in a *precision versus recall curve*. This curve is calculated by introducing 11 standard recall levels, for which the precision is calculated. This results into a curve, as shown in figure 4.

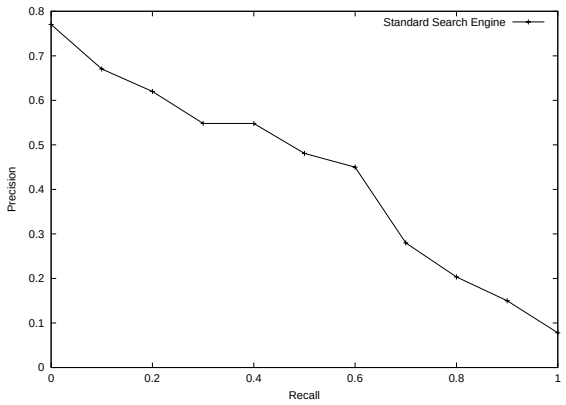

Figure 4: Precision at 11 standard recall levels.

To illustrate how the precision curve at 11 standard recall levels of the figure should be interpreted, consider the following. When the *top* 30% of the documents in the answer set is evaluated, the retrieval strategy has a precision of 0.5480. The graph shows the typical behaviour of IR systems. Increasing the recall of the answer set, will gradually cause a decrease in precision. Interpolation is used to determine the precision at recall level 0.

Alternatively, the average precision at given *document cut-off values* can be computed. This approach provides additional information with respect to the retrieval performance of the ranking algorithm. The result is a curve, which shows the average precision of the retrieval strategy when 5, 10, ..., 100 documents are seen.

The recall-precision curves explained in this section are usually not computed over a single query, but over the entire set of information requests. This provides a good estimate of the overall performance of a retrieval strategy. In [BYRN99] it is pointed out that it is equally important to investigate the behaviour of a retrieval strategy for the individual queries.

Two reasons are brought up. First, the average precision might disguise important anomalies of the retrieval strategy. Second, when two retrieval strategies are compared it is interesting to seen whether one of them outperforms the other for each query. In literature evaluation measures that provide this kind of information are referred to as *single value summaries.*

The *average R-precision* is an example of such a measure. A single value of the ranking is computed by taking the precision at the R-th position in the ranking, where R is the total number of relevant documents for the current query. These measures can be plotted in R-precision histograms using the following rule to compare the retrieval history of the retrieval strategies A and B. Let $RP_A(i)$ and $RP_B(i)$ be the R-precision values of the corresponding retrieval strategies for the i-th query. The difference between these strategies is then simply calculated as:

$$RP_{A/B}(i) = RP_A(i) - RP_B(i)$$

An example R-precision histogram is given in figure 5. A positive precision means that strategy A outperformed strategy B and vice versa if the precision is negative.

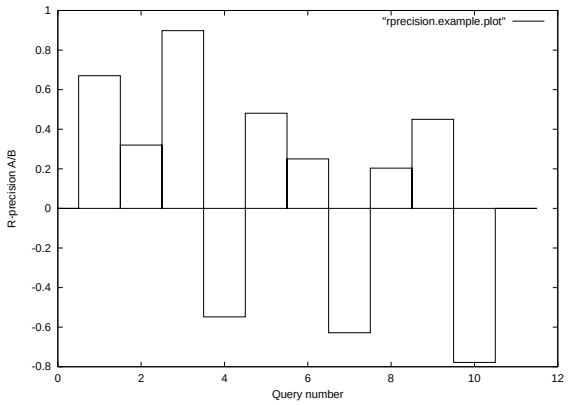

Figure 5: R-precision histogram, comparing the retrieval strategies A and B

Other measures which are used to emphasis certain aspects of the retrieval performance evaluation are discussed in [BYRN99, DF98].

4.2 Text REtrieval Conference

The Text REtrieval Conference, or shortly TREC, is an initiative of the National Institute of Standards and Technology (NIST) and the Information Technology Office of the Defense Advanced Research Projects Agency (DARPA). The goal of the conference is 'to encourage research in information retrieval from large text applications by providing a large test collection, uniform scoring procedures, and a forum for organisations interested in comparing their results'.

The ad-hoc task of the TREC test reference collection uses the following four basic types of evaluation measures: (1) *summary table statistics*, (2) *recall-precision averages*, (3) *document level averages*, and (4) *average precision histogram*

5 Experimental setup

Evaluation of retrieval strategies for information retrieval used to be a difficult task, since good benchmarks and large test collections were missing. But today, the TREC collection among others, provides good experimental platforms for most IR systems. Unfortunately, the techniques used for the Webspace Method cannot be applied on such test collections, due to the requirements of the conceptual model. To evaluate the

retrieval performance of the Webspace Method a new test collection is built, which satisfies the conceptual constraints of the Webspace Method. For this purpose the document collection originally used for the 'Lonely Planet' case-study is used.

5.1 'Lonely Planet' test reference collection

Like the test reference collections described in Section 4 the 'Lonely Planet test reference collection consists of a collection of documents, a set of information requests (topics), and a set of relevant documents for each information request, provided by specialists.

- **Document collection**. The Lonely Planet document collection consists of approximately 6500 documents, which describe destinations all over the world. For each destination, several documents exist, containing items like 'history', 'culture', 'activities', 'facts', and many other destination related information. A large subset of the document collection is formed by the postcard collection. A postcard contains some information about a destination, which is send in by a traveller, containing personal experiences with respect to the destination. Furthermore, for nearly each destination a slideshow is available, containing a series of pictures, which illustrate the typical characteristics of that destination. To get an impression of the document collection, please visit the original website, located at `http://www.lonelyplanet.com/`.
- **Topics**. The test collection contains 20 topics (queries) related to the Lonely Planet document collection. In Table 1 the description of such a topic is given.

TOPIC 5:
Search for cultural information about destinations, where the description contains terms like:"skin colour racism politics church".

Query terms	*Stemmed terms*
culture skin colour racism politics church	skin colour racism polit church

Table 1: Description of Topic 5.

- **Relevance judgements**. The relevance judgements, i.e. the set with relevant documents, for each information request (topic), is determined by a *blind review pooling method*. For each topic, the set of documents found by the different search engines were collected and mixed. Twenty experts were asked to evaluate the (combined) retrieved document set for each topic.

5.2 Evaluation measures

The primary evaluation of the experiment is based on the evaluation measures used for TREC. As discussed in Section 4.2, four classes of evaluation measures are used: (1) the summary table statistics, (2) recall-precision averages, (3) document level averages, and (4) the average precision histogram. In the next section a discussion of the results of the experiments is given, based on this classification.

5.3 Runs

Last but not least, three runs were carried out on the 'Lonely Planet' test collection. Below a short description of each run (search engine) is given. For each topic, only the fifteen highest ranked documents were evaluated as the answer set of the search engine for a given topic. As a result the experts have to

examine a set of documents, with the minimal size of 15 documents, if the three search engine produce the same answer set for a topic, and a maximum of 45 documents, if the answer sets are completely disjoint.

- **Standard search engine**. The standard search engine (SSE) is based on the well-known vector-space model. This model forms the basis for commonly used search engines on the WWW, like Google and Alta-Vista. Its interface allows a sequence of query terms to be formulated, which is used to find a ranking of relevant documents. For each topic, a set of query terms is defined, which represents that topic. The query terms for Topic 5 are given in Table 1.

- **Webspace search engine**. The webspace search engine (WSE), is of course based on the Webspace Method, and uses the webspace schema for formulation of the queries (topics) over the document collection. The text retrieval component of the WSE is based on exactly the same vector-space model as used for the standard search engine.

  Instead of indexing the entire document, only the Hypertext-fragments are indexed by the TextDaemon. The stemmed terms, given in Table 1 are the (stemmed) terms used by the WSE, to evaluate the relevance of the given Hypertext-objects. To be able to make a comparison between the search engines, the WSE only returns the document URLs containing the relevant information, instead of using the option of composing user-defined views.

- **Fragmented webspace search engine**. The fragmented webspace search engine (FWSE) is a variant of the webspace search engine, which uses a horizontally fragmented TextDaemon.

  Instead of building one term index for all Hypertext-objects, a separate term index is built for each conceptual attribute of type Hypertext. For instance, in case of the webspace schema presented in Figure 1, three different term indexes are built using the triggers: (1) Postcard.message.Hypertext, (2) Destination.introduction.Hypertext, and (3) InfoItem.description.Hypertext. The motivation behind the fragmentation is the following:

  a) The Webspace Method introduces a conceptual index, which exploits the *semantical structure* of a document. Thus the textual descriptions associated with a concept, will probably also contain semantically different terms.
  b) If no fragmentation is used, the less frequently occurring index terms related to a specific conceptual class and attribute have a relatively low *tfidf* if the same terms occur frequently in the Hypertext-objects associated with a different class and attribute. The fragmentation causes a correction in the *tfidf*.
  c) This correction is only useful, if the query uses more than one query terms when searching *Hypertext*-fragments, because a difference in the final ranking of the *Hypertext*-objects, can only occur if the difference between the *idf*-values of the terms, with and without fragmentation is large enough. The more terms are specified, the more likely it is that a difference in ranking will occur. In [KW01] some tests, also based on the 'Lonely Planet' case-study, are described, which provide detailed information on the implementation issues, and show that there is actually a change in ranking, when comparing the results of the FWSE with the WSE.

6 Experimental results

The evaluation method provides statistics, which to a certain degree, are comparable with trends obtained from the statistical results from TREC. Below the four basic measures are discussed.

6.1 Summary table statistics

In Table 2 the *summary table statistics* for the three runs of the experiment are given. They provide a first indication of the performance of the different search engines.

Total number of documents over all queries			
	SSE	**WSE**	**FWSE**
Answer set (A)	300	287	287
Relevant document set (R)	222	222	222
Relevant doc. in answer set(R_a)	104	187	188

Table 2: Summary table statistics.

The first row of the table provides information about the size of the answer set containing the retrieved documents for each of the search engines. The answer set of the SSE contains the maximum of 300 documents over the 20 given topics, while both the webspace search engines did not always return 15 relevant documents, due to the conceptual constraints of the query. This has a positive influence on the precision. But reduces the recall, if the webspace search engines do not find all the relevant documents.

The total number of relevant documents (R), over the entire set of documents, for all topics is 222. The third row shows the set R_a, which gives a first indication of the performance of the search engines. The standard search engine found 104 relevant documents, where the webspace search engines have found 187 and 188 documents. This is almost twice the amount of relevant documents found by the standard search engine.

6.2 Recall-precision averages

The summary table statistics only supply average statistics of a run, due to their set-based nature. More detailed information on the retrieval performance is normally expressed in precision versus recall. This also takes into account the evaluation of the ranking. Figure 6 shows the recall-precision curve. This curve computes the average performance over a set of topics, interpolated over 11 standard recall levels (0.0 to 1.0, with increments of 0.1). At standard recall level i, it shows the maximum precision of any recall level greater than or equal to i. The optimal IR system, will have a curve, equal to a horizontal line, at a precision of 1.

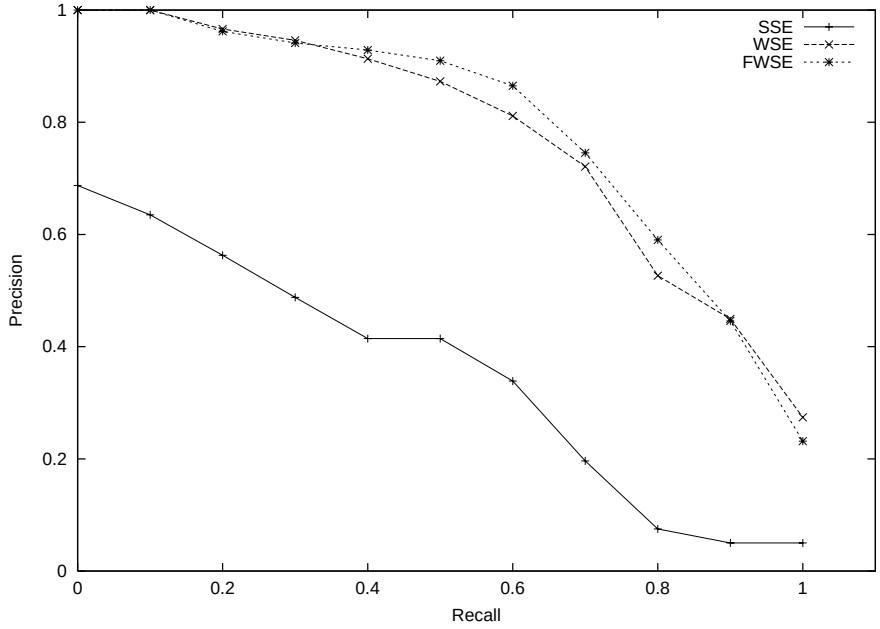

Figure 6: Interpolated recall-precision curve.

The recall-precision curve of the standard search engine starts at a precision of 0.69 and ends with a

precision of 0.05 (see Table 3). Both webspace search engines start at the optimal precision (1.00), and end with a precision of 0.27, and 0.23. The resulting curves show that the performance of the search engines that use the Webspace Method have a much higher performance, than the standard search engine, using the same IR model. There is only a small increase in performance when the fragmented IR model is used in combination with the Webspace Method. This small increase in performance is caused by a better ranking of the relevant documents, rather than that different documents were found.

Recall versus precision				
Run	Average Precision	Initial Precision	Precision @ 100	Recall
SSE	0.3389	0.6873	0.0500	0.4684
WSE	0.7687	1.0000	0.2740	0.8423
FWSE	0.7806	1.0000	0.2316	0.8468

Table 3: Precision measures.

The values for the *non-interpolated average precision* for the three search engines show that the WSE and the FWSE, with their precisions of 0.7687 and 0.7806, respectively, cause a large improvement in the precision by almost a factor 2. At the same time, the average recall of the webspace search engines is also increased by a factor 1.8, compared to the standard search engine. Note that the non-interpolated average precision measure especially rewards systems, that rank relevant documents at a high position.

From these measures it is clear that the schema-based approach for querying document collections, in combination with the integration with information retrieval, as introduced by the Webspace Method, is responsible for the increase in retrieval performance.

6.3 Document level averages

The document level averages provide more insight in the quality of the ranking mechanism. Based on pre-determined document cut-off values, the average precision of the search engine is given after seeing N documents. In Table 4 document precision averages are given for all three runs, after retrieving x documents. It reflects the actual measured retrieval performance, from the user's point of view. The document precision average is calculated by summing the precisions at the specified document cut-off value, divided by the number of topics (20). Again the the precision of the webspace search engines is much better than for the standard search engine. The table also shows that the ranking computed by the FWSE is better than for the WSE.

Results of the standard evaluation method			
	SSE	WSE	FWSE
At 5 docs	0.4400	0.8200	0.8500
At 10 docs	0.3650	0.6950	0.7100
At 15 docs	0.3476	0.6233	0.6267
R-precision	0.3686	0.7363	0.7419

Table 4: Document level averages.

The *R-precision* is the precision after R documents have been retrieved. In this case R equals the number of relevant documents. It de-emphasises the effect of the ranking of a run. It shows that the R-precision for both webspace search engines is 0.74.

6.4 Average precision histogram

The average precision histogram of Figure 7 shows the average non-interpolated precision for each of the three runs per topic. This measure gives a good indication of the performance of a run on specific topics.

The measured precisions for the topics 1, 3, 5, 7, 8, 9, 11, 12, 13, 15, 17, 19, and 20 show a huge increase in retrieval performance, when the topic is evaluated using the Webspace Method. In some cases, where the standard search engine performed really poorly, this was caused by the conceptual constraints, as is the case for Topic 15: '*Find postcards, send from destinations within North East Asia containing information about 'taxi shuttles airport or public transportation'.*'. The constraint '*destinations within North East Asia*' cannot be handled by normal search engines.

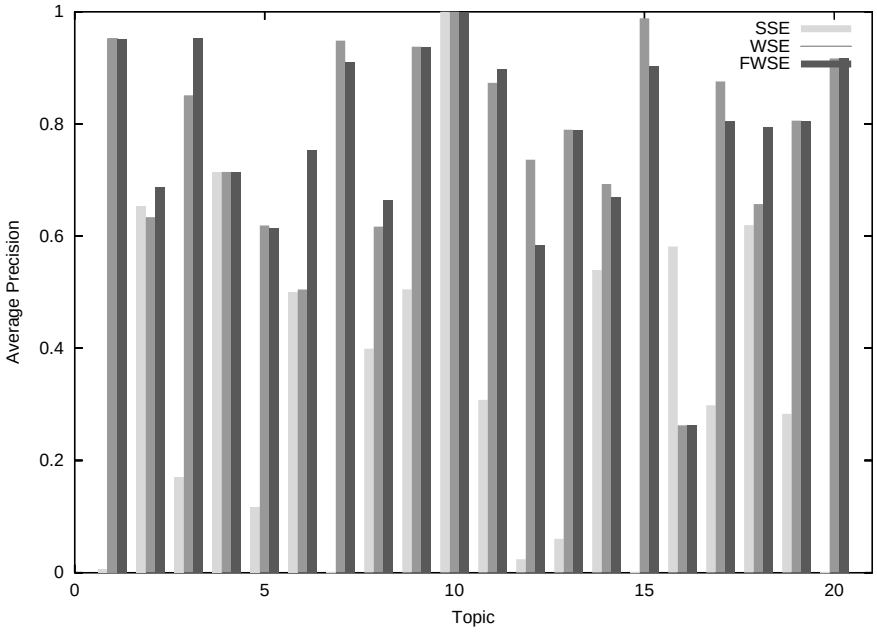

Figure 7: Average precision histogram.

Only on Topic 16 the standard search engine performed better than the webspace search engines. While the performance of the search engines was nearly the same for topics 2, 4, 10, and 18.

When comparing both webspace search engines, it turns out that the performance of the FWSE is the same, or slightly better for most queries than the WSE, with the exception of topics 12, 14, 15, and 17. While for topics 2, 3, 6, 8, and 18 the ranking produced by the FWSE is clearly better, than the one produced by the WSE.

7 Conclusions

The results of the retrieval performance experiment show a huge increase in performance, when searching for documents, using the Webspace Method. Although the setup and results of the experiment are good and reliable, more experiments should be carried out to validate the conclusions at this point.

For instance, a relatively low document cut-off point (15 documents per topic) is used, compared to the TREC-benchmark. This might lead to in-precise precisions, at the high recall levels (>0.8). On the other hand, the size of the Lonely Planet test collection is also a lot smaller, which eliminates the necessity of evaluating the relevance of documents which are not highly ranked. Therefore, it is expected that the document cut-off point is not chosen too low.

Both the recall and precision values, found for both webspace search engines are approximately a factor two higher, than the values found for the standard search engine. This typically illustrates the impact of the Webspace Method on the retrieval performance. When comparing the results of the standard search engine with the recall-precision curves of the search engines that participated in TREC, a similar trend is found. Unusual are the (extremely) high precision values of the webspace search engines at the lower

recall-levels, caused by the conceptual model of the Webspace Method. These values show the contribution of the Webspace Method to the retrieval process.

From the average precision histogram it can be concluded that on average the fragmented webspace search engine produces a better ranking than the standard webspace search engine. Furthermore, both webspace search engines are clearly capable to answer queries that cannot by answered by the standard search engines which solely have to rely on text retrieval techniques. Especially, if the information requested, is shattered over more that one document.

References

[AO98] G. O. Arocena and Mendelzon O. WebOQL: Exploiting document structure in web queries. In *proceedings of the International Conference on Data Engineering (ICDE)*, pages 24–33, 1998.

[BdVBA01] H.E. Blok, A.P. de Vries, H.M. Blanken, and P.M.G. Apers. Experiences with ir top n optimization in a main memory DBMS: Applying 'the database approach' in new domains. In *proceedings of British National Conference of Databases BNCOD*. Springer, July 2001.

[BHC+01] H.E. Blok, D. Hiemstra, S. Choenni, F. de Jong, H.M. Blanken, and P.M.G. Apers. Predicting the cost-quality trade-off for information retrieval queries: Facilitating database design and query optimization. In *proceedings of the International Conference on Information and Knowledge Management*, Atlanta, USA, November 2001.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, 1999. ISBN_ISSN: 0-201-39829-X.

[CCD+99] S. Ceri, S. Comai, E. Damiani, P. Fraternali, S. Paraboschi, and L. Tanca. XML-GL: a graphical language for querying and restructuring XML documents. In *proceedings of the International World Wide Web Conference (WWW*, pages 1171–1187, Canada, 1999.

[CFR+01] D. Chamberlin, D. Florescu, J. Robie, J. Simeon, and M. Stefanescu. XQuery: A query language for XML. Technical report, World Wide Web Consortium (W3C), `http://www.w3.org/TR/xquery`, Februari 2001.

[DF98] D.A.Grossman and O. Frieder. *Information Retrieval: Algorithms and Heuristics*. Kluwer international series in engineering and computer science. Kluwer Academic Publishers, 1998. ISSN_ISBN: 0-7923-8271-4.

[dV99] A.P. de Vries. *Content and Multimedia Database Managements Systems*. PhD thesis, Centre for Telematics and Information technology, Enschede, the Netherland, December 1999.

[dVED98] A.P. de Vries, B. Eberman, and D.E. Kovalin D.E. The design and implemantation of an infra structure for multimedia digital libraries. In *proceedings of the International Database Engineering & Applications Symposium*, pages 103–110, Cardiff, UK, 1998.

[dVW99] A.P. de Vries and A.N. Wilschut. On the integration of IR and databases. In *proceedings of the IFIP 2.6 Working Conference on Data Semantics 8*, 1999.

[FG00] N. Fuhr and K. Grossjohan. XIRQL: An extension of XQL for information retrieval. In *proceeding of ACM SIGIR Workshop On XML and Information Retrieval*, Athens, Greece, July 2000.

[HTK00] Y. Hayashi, J. Tomita, and G. Kikui. Searching text-rich xml documents with relevance ranking. In *proceedings of the ACM SIGIR 2000 Workshop on XML and Information Retrieval*, Athens, Greece, July 2000.

[KP96] W. Kraaij and R. Pohlmann. Viewing stemming as recall enhancement. In *proceedings of the 19th Annual International ACM Conference on Research and Development in Information Retrieval SIGIR*, pages 40–48, 1996.

[KW01] I.A.G.H. Klerkx and W.G.Tijhuis. Concept-based search and content-based information retrieval. Master's thesis, Saxion Hogeschool Enschede, in cooperation with the department of Computer Science, University of Twente, Enschede, The Netherlands, march 2001. (in Dutch).

[MMA99] G. Mecca, P. Merialdo, and P. Atzeni. Araneus in the era of xml. *IEEE Data Engineering Bullettin, Special Issue on XML*, September 1999.

[MMM97] Alberto O. Mendelzon, George A. Mihaila, and Tova Milo. Querying the world wide web. *Int. Journal on Digital Libraries*, 1(1):54–67, 1997.

[Por80] M. F. Porter. An algorithm for suffix stripping. *Program*, 14(3):130–137, 1980. Belfast, Northern Ireland.

[Pub01] Lonely Planet Publications. Lonely planet online. `http://www.lonelyplanet.com/`, march 2001.

[SYW75] G. Salton, C. Yang, and A. Wong. A vector-space model for automatic indexing. *Communications of the ACM*, 18(11):613–620, 1975.

[vZA99] R. van Zwol and P.M.G. Apers. Searching documents on the intranet. In *proceedings of Workshop on Organizing Webspace (WOWS'99), in conjunction with Digital Libraries 1999*, Berkeley (CA), USA, August 1999.

[vZA00a] R. van Zwol and P.M.G. Apers. Using webspaces to model document collections on the web. In *proceedings of Workshop on WWW and Conceptual Modelling (WCM'00), in conjunction with ER'00*, Salt Lake City (USA), October 2000.

[vZA00b] R. van Zwol and P.M.G. Apers. The webspace method: On the integration of database technology with information retrieval. In *proceedings of Ninth International Conference on Information and Knowledge Management (CIKM'00)*, Washington DC., USA, November 2000.