Medical Informatics Europe '99
P. Kokol et al. (Eds.)
IOS Press, 1999

WWW search engine for Slovenian and English medical documents

Jure Dimec[*], Sašo Džeroski[**], Ljupčo Todorovski[*], Dimitar Hristovski[*]
[*]*Institute of Biomedical Informatics, Faculty of Medicine, Vrazov trg 2, 1105 Ljubljana, Slovenia*
{jure.dimec | ljupco.todorovski | dimitar.hristovski}@mf.uni-lj.si
[**]*Institute Jožef Stefan, Jamova 39, 1111 Ljubljana, Slovenia*
saso.dzeroski@ijs.si

Abstract. The information tool for the organization and searching of Slovenian and English medical documents is presented. The tool, partly still in development phases, performs automatic subject description of documents, searching with natural language queries and ranking of search hits according to their relevance. The search engine allows the searcher to use relevance feedback in order to perform incremental improvement of search results.
The machine learning system TILDE for learning user profiles was also applied. Documents marked by the user as relevant or non-relevant are used to find characteristics that distinguish relevant documents from non-relevant ones.

1. Introduction

World Wide Web (WWW) is increasingly useful as a source of information in research and development. In the flood of documents aimed at different user populations the idea of digital library is steadily gaining importance. Digital library *sensu stricto* could be described as a collection of electronic documents accessible on network on which some sort of quality control was performed and as a suite of information tools for discovery and presentation of these documents. Among information tools, search engines are the most important.

Digital libraries in their present-day rudimentary forms only partially solve the problems rooted in the abundance of untrustworthy documents on Internet. Good information tool designed for the demanding user should enable him or her to search through information sources organised in local digital library(es) and, at the same time, offer an individualised discovery and access to the wealth of potentially useful documents elsewhere on WWW.

In this paper we would like to report on our work on both tasks. We are (a) developing the database of automatically built subject descriptions of unstructured and structured documents archived in our digital library, (b) developing a search engine for searching this database with natural language queries, and (c) building the user profiles with descriptions of users' information needs. User profiles built with the machine learning methodology will automate the discovery of relevant documents on WWW. With respect to documents published, medicine in Slovenia is predominantly bilingual, Slovenian and English languages being almost equally important, therefore all language-dependent procedures are dealing with these languages.

The paper begins with the description of automatic indexing. The third and fourth sections are devoted to the search engine and to the arguments for developing it from the scratch. The fifth section presents preliminary experiments in building user profiles with machine learning system TILDE.

2. Database with subject descriptions of Slovenian and English medical documents

Dynamic nature and number of information sources on Internet make indexing with keywords or subject headings attributed by human experts virtually impossible. The task is realisable only with automatic indexing – the approach used for the construction of databases of famous search services like AltaVista, Excite, Infoseek, and others. With some simplification, the automatic indexing could be described as a three-step procedure:

1. using of stop-words list to eliminate the words with minimal informational value;
2. stemming to normalise the different forms of the same word to the same stem; and
3. weighting of the remaining stems to reflect their informational value.

We use all three steps for documents in both languages but only processing of Slovenian documents is described here. For automatic indexing of English documents we use methods often described in literature [e.g. 1].

At the time of reporting the pilot database installation consists of documents from electronic versions of two medical journals: Slovenian edition of JAMA (Journal of the American Medical Association, http://www.mf.uni-lj.si/jama/jama-slo-e.html) and ISIS (Journal of Slovenian Chamber of Medicine, http://www.mf.uni-lj.si/isis/isis.html) and bibliographic records with abstracts from national database Biomedicina Slovenica (http://www.mf.uni-lj.si/cgi-bin/wow/bs_frm?lang=ENG).

2.1. Stop-words and stemming

Each word from document to some extent represents the document's subject and is a candidate for being an index term. Words called stop-words are exceptions. Stop-words are words that are evenly distributed in document base and as a consequence bear the least amount of information in documents. They mainly belong to several word groups like propositions, adverbs, pronouns, etc. Slovenian stop-words list presently contains near 1600 words and word forms, which is much compared to typical English lists with 250 – 400 words. Words found in list are excluded from further processing.

The quality of stemming, which is also language-dependent procedure, is a critical factor and the quality of both automatic indexing and searching depends on it to the great extent. With stemming we try to determine the string of characters that represents all forms of the word and discriminates them from other words. Often, but not always, the stem corresponds to the word root. While reports exist on little or medium importance of stemming of English documents it is very important in languages with rich morphology. Of such languages the Slovenian is a good example.

We are aware of two previous successful attempts to develop statistical algorithms for stemming of Slovenian texts and used for construction of document databases [2, 3]. Both were based on longest match principle and used extensive lists of word endings, 1205 and 5276 endings, respectively. In longest match algorithms the list of endings is searched for the longest ending that could be mapped to the terminal part of the word and ending is cut off the word. These algorithms often exhibit overstemming producing too short stems and they are less adequate for medical sublanguage due to abundance of words of Greek or Latin origin. In our project we developed new algorithm, which works as a combination of stemming rules and a set of valid stem examples.

We use three groups of the stemming rules firing consecutively: rules for endings that divide the word at the consonant-vowel pair, rules for the remaining consonant-consonant pairs, and recoding rules. The cursor marking the potential division point between stem and ending is gradually moving from right to left, one character at the time. At each character the full stemming is performed (if possible) and if it succeeds and the resulting stem is found in the list of legal stems, the procedure is finished. In the case when the stem is never found in the list of legal stems it enters the temporary list, that have to be occasionally inspected, corrected, and merged with the legal list. This new stemmer, which could be attributed as the shortest match algorithm, is evolutive, self-learning, and is found to be better adapted to the medical sublanguage.

2.2. Weighting of stems

Weight of the stem denotes the share of information that this particular stem contributes to the whole information load of the document. It depends mainly on the stem's frequencies in the document (intradocument frequency) and in the database (interdocument frequency). Basically, the higher intradocument stem frequency means more important aspect of the content the stem is about; and lower interdocument stem frequency results in higher ability of stem to discriminate documents that contain it from others in database. Both measures are combined in stem weight in particular document.

Unavoidable characteristic of Internet is its dynamic state, which prevents the computation of the final stem weights during automatic indexing. Parameters derived from interdocument frequencies could be taken into account only during searching thus reflecting the database state at that moment. Documents in digital library are marked with HTML (Hypertext Markup Language) therefore it is possible to augment stem weights with additional (ad hoc) values of HTML tags. We can expect the word from text body to be less important than the word which is emphasised and this one again less important than the title word.

3. Searching

A good search engine is expected to process queries as natural language sentences and rank resulting pointers to documents according to the computed relevance. Relevance is defined as a measure of similarity between a query and a document. In general it is computed as a sum of stem weights common to the query and the document. In our work we used Croft's probabilistic method [5] supplemented with the value of HTML tags.

All big search engines on WWW use ranking of search hits. Analyses show that the majority of queries are short, two or three words long, causing the quality of ranking that is suboptimal. The searcher is forced to browse through a good part of ranked list (which is often very long) or to change the query and repeat the searching. To avoid this shortcoming we implemented the searching method called relevance feedback, known from the classical document retrieval as one of the methods that give the best search results [4, 5]. The method is based on the interactive dialog with the searcher who provides identifications of relevant documents. Big WWW search services don't use the method because it needs unique identity of searcher or uninterrupted (statefull) search session that is difficult to accomplish with usual CGI programs and HTTP protocol. The course of relevance feedback searching is three-step:

1. the searcher performs first, normal search;
2. receives search hits, inspects several top-ranked documents and mark those that he finds relevant;
3. system automatically reformulates query and performs new search.

The core of the procedure is query reformulation. The system recalculates weights of all stems from the query using the probabilities of their occurrence in relevant and non-relevant documents. Searching with the reformulated query includes new documents to the hit list and changes the positions of previously found documents so those relevant ones climb the list upwards. After receiving each hit list the searcher repeats the inspection of top ranked and previously uninspected documents and mark the relevant. Steps 2 and 3 are repeated as long as there are new relevant documents near the top of the list or the searcher gives up.

4. Why another search engine?

Public search services like AltaVista, Excite, and some others index the predominant part of WWW (according to their authoring institutions), including pages on Slovenian servers. Then, why are we developing new search engine for which it is clear that it can not substitute the big ones?

Automatic indexing is highly language-dependent procedure determining the success of searching. The *lingua franca* of WWW pages is English and indexing is adapted for that language, although some search engines are able to process queries in limited number of other languages. While using them the documents in Slovenian mainly remain hidden. For the time being we are not aware of any working search engine for Slovenian documents that doesn't avoid language-dependent functions by requiring manual truncation of query words although we are aware of two very promising lines of work. We believe that our system at least partly fills this empty space.

We conceive the database of subject descriptions and pointers to documents, search engine, and modules for personalised document discovery as a tool for the advanced use of digital library of medical documents. We are indexing only documents that undergone some sort of quality control and that are useful for students or in professional and research work in Slovenian medicine. One of the consequences of this commitment is relatively small database.

The amount of stored data and relatively limited size of users' population gives us the possibility to implement functions that big search services still can not afford themselves. It is our firm opinion that good search engine should borrow from the behaviour of each of the two existing information retrieval words – local, possibly client/server systems used with bibliographic databases, and big public search services on WWW. We believe that good and exhaustive search could not be performed in one pass therefore user should be able to evolutively improve search results.

The server, search engine and client were written in Java. The search engine is implemented with TxtIndex4, search engine kernel in Java, provided by Novi Forum [10].

5. Learning relevance of documents

In the context of user interests' profiles, documents marked by the user of the search engine as relevant or non-relevant are used to find characteristics that distinguish relevant documents from non-relevant ones. Given the examples of relevant and non-relevant documents, machine learning methods are used to build a model that can be used to predict the relevance of new documents.

Machine learning methods have been already used for predicting the relevance of documents on the WWW within the WebWatcher project [7] and also for modeling individual user's information interests as profiles within the Personal WebWatcher project [8]. However, none of this methods deals with documents in the Slovenian language.

We used machine learning methods for predicting the relevance of English and Slovenian documents from examples. We used the collection of 770 short documents (abstracts from some leading Slovenian medical journals) – half of them in Slovenian and the other half their English translations. Three human experts marked the documents as relevant or non-relevant with respect to 50 different queries, which we assume correspond to 50 topics of interest. We consider each of them as a learning problem, where the relevant documents are positive and the non-relevant ones are negative examples. Therefore, each learning problem has 770 examples. Two series of experiments were performed. The documents were represented as sets of words in the first, and as sets of stems in the second series.

TILDE [9] machine learning system was used to generate logical decision trees that distinguish relevant documents from non-relevant ones. The trees can in most cases be transformed into decision lists of keywords that indicate relevant documents. The motivation for using TILDE was the fact that it can also use background knowledge, such as MeSH or UMLS, in the learning process. Although we did not use such knowledge in the experiments presented here, we plan to do so in the future.

Consider query number 10: "Nastanek, diagnostika in zdravljenje ulkusa, še posebej razjede želodca in dvanajstnika" (in Slovenian) or "The origin, diagnosis and treatment of ulcer, especially duodenal and gastric ulcerations" (in English). From the examples for the query (in the set of 770 documents, only 8 are marked as relevant), TILDE generates the following decision tree:

```
ulcer ?
+--yes: rel_10
+--no: ulkusa ?
   +--yes: rel_10
   +--no: not_rel_10
```

The tree can be interpreted as follows: "If the document includes the word *ulcer*, then it is relevant. The document is also relevant, if it does not include the word *ulcer*, but it include the word *ulkusa* (Slovenian for 'of the ulcer'). Otherwise the document is non-relevant."

In the case of query number 14 ("Kirurško (operativno) zdravljenje zlomov kosti" or "Surgical (operative) treatment of bone fractures") TILDE generated a tree which can be interpreted as follows: "The document is relevant, if it includes one of the words: *fracture*, *zlome*, *calcaneal*, *zlomih* or *zlomov*." The tree misclassifies one of the relevant documents as non-relevant. Note here that Slovene is a highly inflected language: *zlome*, *zlomih* and *zlomov* are all plural forms of the word *zlom* (fracture).

Finally, consider query number 31: "Uporaba ultrazvoka v diagnostiki" or "Use of ultrasound in diagnosis". There was 22 relevant documents for this query out of all 770 documents in the collection. The tree generated by TILDE can be interpreted as follows. A document is relevant, if it includes one of the words *ultrazvočni*, *sonography*, *ultrazvokom*, *ultrasound*, *echocardiographic*, *ehokardiografija*, *hoechst*, *laser* or *ultrazvočno*. Otherwise it is non-relevant. The tree misclassifies three relevant documents as non-relevant and two non-relevant documents as relevant.

In the decision tree for query number 14 the words *zlome*, *zlomih* and *zlomov* appeared. All of them have the same stem *zlom*. Also, in the tree for the query number 31 three words with the stem *ultrazv* appeared (*ultrazvočni*, *ultrazvokom*, *ultrazvočno*). This motivated us to use stemmed versions of the documents, which would probably produce more concise trees/decision lists distinguishing between relevant and non-relevant documents.

When using a set of stems instead of words the decision tree for the query number 14 is: "A document is relevant, if it includes the stem *zlom* or the stem *calcaneu*. Otherwise, the relevance depends on the stem *fractur*: if the stem *fractur* appears and stems *bas* and *manag* do not appear, the document is relevant, else the document is relevant, if it icludes stems *spin* and *oper*, else it is non-relevant. The tree based on stems misclassifies one non-relevant document as relevant.

For the query number 31 TILDE generated the following tree. A document is relevant, if it includes at least one of the stems *ultrazvoč*, *digit*, *ultrasound*, *echocardiograph*, *sonograph*, *ehokardiograf* or *lh*. If the document includes stem *cist*, then it is relevant in case when it does not include stem *premer*. The tree misclassifies two non-relevant documents as relevant.

As we expected, the decision trees based on stems distinguish relevant documents from non-relevant ones better. In the two example queries described above, the misclassification error is smaller when using stems instead of words. Furthermore, none of the relevant documents was misclassified as non-relevant which was not the case with the trees based on words. This is very important, because there are few relevant documents for each query.

Finally, we used some of the learned profiles to generate queries for WWW search engines (e.g. AltaVista). The profiles generated from the set of words abstract representation were used here. The keywords appearing in the profile (identified by the machine learning approach) are used to form a query. The generated queries were submitted to the search engine and the search results inspected manually. Reasonable search results were obtained, especially for profiles on topics with larger numbers of relevant documents.

6. Further work

Having all the necessary elements in place, we will incorporate the above methodology into a search mechanism that will be made available to medical researchers. In addition to the use of machine learning methods for generating user profiles as described in the paper, the system will also include automatically generating queries for different WWW search engines from these profiles. We also plan to incorporate a clustering approach to present the search results of queries generated from profiles. The user will also be able to browse this clustered and ranked list of hits and add more relevant documents to his/her profile.

Acknowledgement

The work presented in this paper was partly supported by the subproject "Intelligent Information Storage and Retrieval of Slovenian and English Medical Documents on Internet" of the INCO COPERNICUS project CRII Cooperative Research in Information Infrastructure, and partly by the project "Digital Library of Slovenian Medical Documents" funded by Slovenian Ministry of Science and Technology.

References

[1] Porter MF. An algorithm for suffix stripping. Program. 14. 1980:130-7.

[2] Dimec J. Computer analysis of Slovenian information language in biomedicine (in Slovenian). M. Sc. thesis. Ljubljana: Medicinska fakulteta, 1989; 77.

[3] Popovič M, Willett P. Processing of documents and queries in a Slovene language free text retrieval system. Literary and Linguistic Computing. 5. 1990:183-90.

[4] Harman D. Relevance feedback and other query modification techniques. In: Frakes WB, Baeza-Yates, editors. Information Retrieval. Data Structures & Algorithms, Englewood Cliffs: Prentice hall, 1992; 241-63

[5] Croft WB. Experiments with representation in a document retrieval system. Research and Development in Information Technology, 1983; 2(1)1-21.

[6] Dimec J. Analysis of information content in different types of Slovenian medical texts and their searching with non-Boolean methods (in Slovenian). Ph. D. thesis. Ljubljana: Medicinska fakulteta, 1995; 108.

[7] Joachims T, Freitag D, Mitchell TM. Web Watcher: A tour guide for the WWW. Proc. IJCAI-97, 15th Intl. Joint Conference on Artificial Intelligence, 1997.

[8] Mladenič D. Personal WebWatcher: Implementation and design. Work report IJS-DP 7472, 1996. Ljubljana, Institute Jožef Stefan.

[9] Blockeel H, De Raedt L. Experiments with Top-down Induction of Logical Decision Trees. Artificial Intelligence, 1998 (in print).

[10] Novi Forum d.o.o. URL: http://www.noviforum.si/