Report on the TREC-2003 Experiment: Genomic and Web Searches

Jacques Savoy, Yves Rasolofo, Laura Perret

Institut interfacultaire d'informatique, University of Neuchatel (Switzerland)

Summary

This year we took part in the genomic information retrieval and information extraction tasks, as well as the named page and topic distillation searches. In carrying out the last two tasks, we made use of link anchor information and document content in order to construct Web page representatives. This type of document representation uses multi-vectors to highlight the importance of both hyperlink and document content.

Introduction

As a part of the TREC-2003 evaluation campaign, the UniNE group is taking part in the genomics and Web tracks. The first section of this paper describes the IR models we used in the genomics information retrieval. Section 2 describes our various approaches to automatically extracting gene descriptions from a given scientific paper, using only its title and its abstract. Section 3 describes our procedures for indexing and retrieving Web pages, based on three document representations, and our distributed indexing framework based on the Okapi probabilistic model. Section 4 explains the IR approach we used to combine both Web page content and anchor information when searching for specific named pages and homepages. Finally, Section 5 describes how our IR scheme can be used within the context of topic distillation tasks.

In order to evaluate our hypothesis, we used the SMART system as a testbed, implementing various vector-space IR schemes and probabilistic models. This year our experiments were conducted on an Intel Pentium III/600 (memory: 1 GB, swap: 2 GB, disk: 6 x 35 GB) and all experiments were fully automated.

1. Genomics Information Retrieval

In carrying out our genomic ad hoc search task, we worked with a Medline collection subset containing 525,938 documents, or more precisely, bibliographic records mainly containing an article title, an abstract and some manually assigned descriptors. The available queries consisted of a gene name (or more precisely, its official name and symbol, together with various alias, associated products and proteins).

This information retrieval task is comparable to the Amaryllis corpus composed of 148,688 scientific bibliographic records written in French. Each of the Amaryllis collection records mainly consisted of an article title, an abstract and some manually assigned descriptors. This collection was included in the CLEF-2002 evaluation campaign (Savoy, 2003).

The indexing procedure we used in both genomic tracks is described in Section 1.1, while Section 1.2 provides an overall description of various IR models. Section 1.3 describes the pseudo-relevance feedback (or blind query expansion) used in our experiments. Section 1.4 shows how we combine the various result lists generated, using different indexing and searching schemes in order to process the same document collection (data fusion). The last section evaluates the retrieval effectiveness achieved by diverse IR models and also that of various combined approaches.

1.1. Indexing Procedure

We chose the SMART system as effective means of searching the Medline collection subset. From the original documents and during the indexing process, we retained only the following logical sections: TI (title), TT (transliterated title from non-Roman alphabet language), AB (abstract), MH (MeSH terms based on the NLM's controlled vocabulary), GS (Gene symbol or abbreviated gene names), and RN (EC/RN numbers are assigned by the Enzyme Commission to designate a particular enzyme).

Only the available descriptions were used to formulate the queries. The "official symbol" field was repeated three times however in order to assign more importance to this particular field.

To form an indexing word, our system takes letter and digit sequences into account, or when preceded and followed by a letter, the following character is used: '.@!_ (the default SMART system list). Thus the strings "IBM360", "U.S", or "sym_name" are viewed as single indexing terms. To this set, we might add / and – characters (a set labeled as "Separator+") in order to view the sequence "DEAD/H" as one indexing term.

Moreover, a stemming procedure was often used in order to reduce to the same root (or stem) inflectional and derivational variants of words. For example, the words "thinking", "thinkers" or "thinks" would be reduced to the stem "think". To achieve this, we employed the Lovins' stemmer (Lovins, 1968) based on

a list of over 260 suffixes (the default stemming approach in SMART). On the other hand, we sometimes preferred using a light stemming approach, wherein only the plural form of English words were removed. To evaluate this stemming approach, we adopted the "S stemmer" (Harman, 1991) based on the following rules:

1. If a word ends in «-ies», but not «-eies» or «-aies» then replace «-ies» by «-y»;
2. If a word ends in «-es», but not «-aes», «-ees» or «-oes» then replace «-es» by «-e»;
3. If a word ends in «-s», but not «-us» or «-ss» then remove the «-s».

1.2. Search Models

In order to define a retrieval model, we must specify how the documents and the requests are to be represented (indexing procedure) and how the similarity between document and query surrogates are to be computed. To achieve this and in order to obtain a broader view of the relative merit of various retrieval models, we evaluated the genomic corpus using 11 search models.

As a first approach, we adopted a binary indexing scheme within which each document (or request) was represented by a set of keywords, without any weight. To measure similarity between documents and requests, we counted the number of common terms, computed according to the inner product (retrieval model denoted "doc=bnn, query=bnn" or "bnn-bnn"). For document and query indexing however binary logical restrictions are often too limiting. In order to weight the presence of each indexing term in a document surrogate (or in a query), we sometimes took account of the term occurrence frequency, thus allowing for better term distinction and increasing indexing flexibility (model denoted "doc=nnn, query=nnn" or "nnn-nnn").

Those terms however that did occur very frequently in the collection were very helpful in distinguishing between relevant and non-relevant items. Thus we could count their frequency in the collection, or more precisely the inverse document frequency (denoted by idf_j), resulting in more weight for sparse words and less weight for more frequent ones. This idf value is usually computed as $\ln(n/df_j)$ where n indicates the number of documents in the collection. Moreover, a cosine normalization could prove beneficial and each indexing weight could vary within the range of 0 to 1 (retrieval model notation: "ntc-ntc"). Table A.1 in the Appendix depicts the exact weighting formulation.

Other variants were also created, especially when considering the occurrence of a given term in a document as a rare event. Thus, it could be good practice to assign more importance to the first occurrence of this word as compared to any successive or repeating occurrences. Therefore, the tf component could be computed as 0.5 + 0.5