York University at TREC 2006: Genomics Track

Xiangji Huang[1], Bin Hu[2] and Hashmatullah Rohian[2]
[1]School of Information Technology, York University, Toronto, Ontario, Canada
e-mail: jhuang@yorku.ca
[2]Department of Computer Science & Engineering, York University, Toronto, Ontario, Canada
e-mail: {hubin,cs233185}@cs.yorku.ca

Abstract

Our Genomics experiments mainly focus on addressing four problems in biomedical information retrieval. The four problems are: (1) how to deal with synonyms? (2) how to deal with the frequent use of acronyms? (3) how to deal with homonyms? (4) how to deal with the document-level retrieval, passage-level retrieval and aspect-level retrieval? In particular, we use the automatic query expansion algorithm proposed at TREC 2005 to construct structured queries for document-level retrieval and we also apply several data mining techniques for passage-level retrieval and aspect-level retrieval. The mean average precisions (MAP) for our automatic run "york06ga1" are 0.3365 at the document-level retrieval, 0.0197 at the passage-level retrieval and 0.1084 at the aspect-level retrieval. The evaluation results show that the automatic query expansion algorithm is effective for improving document-level retrieval performance. However, our retrieval performance on passage-level and aspect-level is poor. One possible reason is that we did not follow the TREC 2006 Genomics track protocol regarding the calculation of passage measures correctly. Therefore, we built a completely wrong index for the passage-level retrieval. Since our aspect-level retrieval is based on the results obtained from our passage level retrieval, the results thus obtained are sub-optimal.

1 Introduction

This paper describes the work done by members of York University for the TREC 2006 Genomics track. Our goal of participating in TREC Genomics track this year is to evaluate the Okapi system at document-level, passage-level and aspect-level retrieval in the biomedical domain.

In our 2004's Genomics experiments, we did not incorporate domain expertise and did not use external biomedical resources [4]. Last year, our experiments focused on the following methodologies: (1) We design two new algorithms for biomedical query expansion. (2) We build structured queries based on extended query terms. (3) We use external biomedical resources for further synonym expansion on the manual run. (4) We use an extended stop word set for improving the retrieval performance. This year our experiments mainly focus on using the automatic query expansion algorithm to construct structured queries for document-level retrieval and applying several data mining techniques for passage-level retrieval and aspect-level retrieval.

The test corpus used in this year's Genomics experiments consists of 162,259 full documents in HMTL format from 49 journals with a total size of 12.3GB. The HTML filename is named as the document's PMID followed by the .html extension. There are 28 topics in total this year. Those topics are available at the link "http://ir.ohsu.edu/genomics/data/2006/2006topics.txt".

The rest of the paper is organized as follows. We first give a brief description of the Okapi information retrieval system in Section 2. Then, our proposed ideas are presented in Section 3. Following that, experimental results are provided in Section 4. Finally, the conclusion and future work are given in Section 5.

2 Weighting and Indexing Using Okapi

We used Okapi BSS (Basic Search System) as our main search system. Okapi is an information retrieval system based on the probability model of Robertson and Sparck Jones [6]. The retrieval documents are ranked in the order of their probabilities of relevance to the query. Search term is assigned weight based on its within-document term frequency and query term frequency. The weighting function used is BM25 [2].

$$w = \frac{(k_1+1)*tf}{K+tf} * \log \frac{(r+0.5)/(R-r+0.5)}{(n-r+0.5)/(N-n-R+r+0.5)} * \frac{(k_3+1)*qtf}{k_3+qtf} \ \oplus \ k_2 * nq * \frac{(avdl-dl)}{(avdl+dl)} \quad (1)$$

where N is the number of indexed documents in the collection, n is the number of documents containing a specific term, R is the number of documents known to be relevant to a specific topic, r is the number of relevant documents containing the term, tf is within-document term frequency, qtf is within-query term frequency, dl is the length of the document, $avdl$ is the average document length, nq is the number of query terms, the k_is are tuning constants (which depend on the database and possibly on the nature of the queries and are empirically determined), K equals to $k_1 * ((1-b) + b * dl/avdl)$, and $\oplus$ indicates that its following component is added only once per document, rather than for each term.

In our experiments, the values of k_1, k_2, k_3 and b in the BM25 function are set to be 1.4, 0, 8 and 0.55 respectively. Our system also supports the structured queries for searching. That is: several different terms that are connected by '+' sign can be used to represent the same concept. For example, "COPII+COP2" stands for "COPII" and "COP2" are synonym.

We use the Okapi system to build the genomics index. In the experiments, all the hyphens have been replaced by the space sign. We also used the standard stop word set provided by Okapi for building the indexing.

3 Our Methods

In the following, we will focus on presenting the algorithm we use for automatic query expansion and the data mining techniques we use for passage and aspect retrieval in our experiments.

3.1 Automatic Query Expansion Algorithm

Information Retrieval in the context of biomedical databases has the following three major problems [3]: the frequent use of (possibly non-standardized) acronyms, the presence of homonyms (the same word referring to two or more different entities) and synonyms (two or more words referring to the same entity). How to deal with an abundant number of lexical variants of the same term is a challenging task in biomedical IR. This year we address these problems by using the following automatic query expansion algorithm.

Before we present this algorithm, we would like to define the following two terms: "break-point" and "replacement". A break-point is a position in a string that can be broken into two parts separated by a space. It can be (1) a hyphen; (2) a position between two letters which have different cases except for the first and second positions of a word; (3) a position between a letter and a digit. For example, the word "185delAG" has 2 break-points. Thus, its variants are "185 delAG", "185del AG" and "185 del AG". A replacement is a substring in a string that can be replaced by a different string and the string after replacing still represents the same meaning as the original one. For example, the number "2" in "COP2" is a replacement which can be replaced by "ii". "alpha" is a replacement that can be replaced by "a" and "beta" is a replacement that can be replaced by "b", and so on. Given these two definitions, The $Algorithm$ 1 is described in Figure 1.

3.2 Passage-Level and Aspect-Level Retrieval

For the TREC 2006 Genomics track, a new task was developed that focuses on passage-level and aspect-level retrieval. "Aspect" is defined similar to how it was defined in the TREC Interactive Track aspectual recall task, representing all answers that are similar.

Input: a name of gene with the break-point from TREC topics
Output: a list of possible variants for the gene name

Method:
(1) Locate all the break-points
(2) For the first break-point, create the variants and put them into the set K.
Repeat this process until all the break-points have been processed
(3) For each variant generated from Step 1 and 2, substitute the replacement with
the corresponding string for creating more variants

Figure 1: The algorithm for the automatic query expansion

Run	Description	Measure (MAP)		
		Document	Aspect	Passage
$york06ga1$	automatic	0.337	0.108	0.0196
$york06ga3$	automatic	0.327	0.103	0.0187
$york06ga4$	automatic	0.344	0.096	0.0135

Table 1: Official results at the 2006 Genomics track

For the passage-level retrieval of each topic, we search against both document level and passage level indexes, and then combine the results from both indexes. For the aspect-level retrieval of each topic, only unsupervised methods are used since we have no training data and no lead on how to classify the retrieved passages. Therefore, the following three steps are used for the aspect-level retrieval. First, an unsupervised method was used to convert the passage content into a vector of words. Second, an unsupervised attribute selection method was used to select important words. Third, a simple K-Mean EM method was used to do the aspect classification. Implementations of those data mining algorithms came from the standard Weka package [7].

4 Experiments

Our experiments were conducted on a double-processor server which has 2 Intel Xeon 2.40GHz CPU and 2G memory. The version of Linux kernel we used is version 2.4.26. York University submitted three runs in total for the 2006 TREC Genomics track.

Figure 2 depicts the overall flowchart of our search system for the 2006 Genomics track. "Run 1" in Figure 2 stands for the result without being classified, which we use as the submission for the passage-level retrieval. "Run 2" in Figure 2 stands for the result classified by using data mining techniques, which we use as the submission for the aspect-level retrieval. "Run 3" is the sorted result for the document-level retrieval.

The mean average precisions (MAP) for our automatic run "york06ga1" are 0.3365 at the document-level retrieval, 0.0197 at the passage-level retrieval and 0.1084 at the aspect-level retrieval. Detailed results are presented in Table 1. Performance comparisons in terms of number of relevant documents retrieved, MAP and R-precision for document-level retrieval are shown in Table 2.

Run	Description	Best	$>=$ Median	$<$ Median
york06ga1	automatic	1	17	8
york06ga3	automatic	0	17	9
york06ga4	automatic	1	16	9

Table 2: Performance comparison of 28 topics on the 2006 Genomics datasets

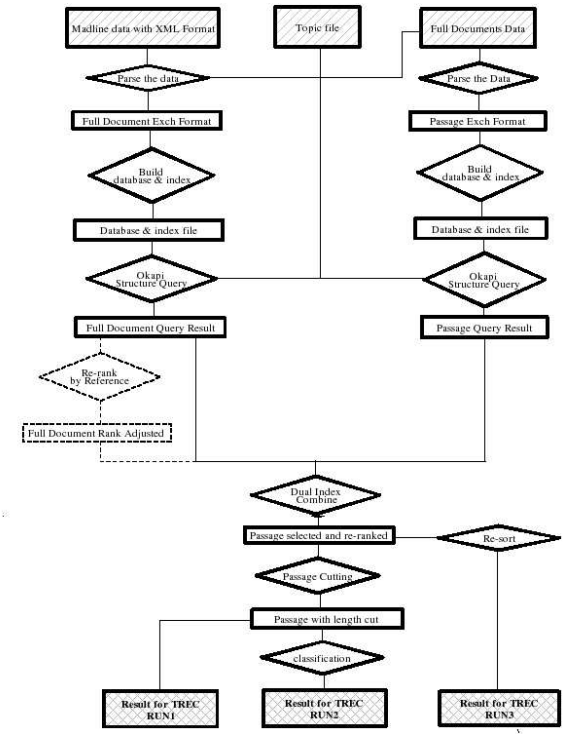

Figure 2: System Architecture

As we can see from Table 1, the retrieval performance on passage-level is poor. One possible reason is that we did not follow the TREC 2006 Genomics track protocol regarding the calculation of passage measures correctly. Because of this, a wrong index was built for the passage-level retrieval. This index was built on the basis of paragraphs appearing in the documents. However, the evaluation for the 2006 genomics track is based on the sentences instead of the paragraphs. For example, there is a paragraph which contains six relevant sentences for "Topic 160". Therefore, there are six relevant "passages" or "records" inside this paragraph. In our experiments, even if our retrieval system could retrieve this long paragraph with all the six relevant passages, we could only get a low grade because of retrieving this long paragraph instead of retrieving six different relevant sentences.

5 Conclusions and Future Work

The contributions of our work are as follows. First, we have designed and implemented an automatic query expansion algorithm. This algorithm is simple, easy to implement and do not need any external biomedical resource. We find that it can improve the retrieval performance on document-level retrieval in the biomedical domain. Secondly, we investigate how to apply data mining techniques for passage-level and aspect-level retrieval in the biomedical domain. Our future work include: (1) building a sentence-based index (2) applying data mining techniques for passage-level and aspect-level retrieval and (3) conducting extensive experiments for evaluation.

6 Acknowledgements

This research is supported in part by the research grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada. We also would like to thank Luo Si from Purdue University and Miao Wen for their help.

References

[1] The Medstract Project: AcroMed 1.1. URL address: http://medstract.med.tufts.edu/acro1.1/

[2] M. Beaulieu, M. Gatford, X. Huang, S. E. Robertson, S. Walker and P. Williams (1996), Okapi at TREC-5. *Proceedings of 5th Text REtrieval Conference*, pp. 143-166, 1996.

[3] Stefan Buttcher, Charles L. A. Clarke, Gordon V. Cormack (2004), Domain-Specific Synonym Expansion and Validation for Biomedical Information Retrieval (MultiText Experiments for TREC 2004). *Proceedings of the 13th Text Retrieval Conference*, 2004.

[4] X. Huang, Y. R. Huang, M. Wen and M. Zhong (2004). York University at TREC-13: HARD and Genomics Tracks. *Proceedings of the 13th Text Retrieval Conference*, 2004.

[5] LocusLink. URL address: http://www.ncbi.nih.gov/locuslink/ *National Center for Biotechnology Information*, 2005.

[6] S. E. Robertson, J. K. Sparck. Relevance Weighting of Search Terms. *Journal of the American Society for Information Science* 27, May-June 1976, p129-146

[7] I. Witten, E.Frank. Data Mining: Practical Machine Learning Tools and Techniques. Morgan Kaufmann, San Francisco, 2005.