York University at TREC 2005: Genomics Track

Xiangji Huang[1], Ming Zhong[2] and Luo Si[3]

[1]School of Information Technology, York University, Toronto, Ontario, Canada

e-mail: jhuang@yorku.ca

[2]Department of Computer Science, York University, Toronto, Ontario, Canada

e-mail: ming@cs.yorku.ca

[3]School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213, USA

e-mail: lsi@cs.cmu.edu

Abstract

Our Genomics experiments mainly focus on addressing three major problems in biomedical information retrieval. The three problems are: (1) how to deal with synonyms? (2) how to deal with the frequent use of acronyms? (3) how to deal with homonyms? In particular, we propose two query expansion algorithms to construct structured queries for our experiments. The mean average precision (MAP) for our automatic run "york05ga1" using *Algorithm* 1 was 0.2888 and for our manual run "york05gm1" using *Algorithm* 2 was 0.3020. The evaluation results show that both algorithms are effective for improving retrieval performance. We also find that some other techniques such as pseudo-relevance feedback and using an extended stop word set can make positive contributions to the retrieval performance.

1 Introduction

This paper describes the work done by members at York University and CMU for the TREC 2005 Genomics track. This year we only participated in the Ad hoc retrieval task of the Genomics track. Our goal of participating in TREC Genomics track is to evaluate the Okapi system in the biomedical domain.

In our last year's Genomics experiments, we did not incorporate domain expertise and did not use external biomedical resources [4]. This year our experiments mainly focus 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.

The test corpus used in this year's Genomics experiments consists of 4,591,008 different documents with a total size of 14GB. There are 50 topics which are categorized into 5 templates: (1) Find articles describing standard methods or protocols for doing some sort of experiment or procedure; (2) Find articles describing the role of a gene involved in a given disease; (3) Find articles describing the role of a gene in a specific biological process; (4) Find articles describing interactions (e.g., promote, suppress, inhibit, etc.) between two or more genes in the function of an organ or in a disease; (5) Find articles describing one or more mutations of a given gene and its biological impact.

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].

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

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 extended stop word set instead of just using the standard stop word set provided by Okapi for building the indexing. We found that a small improvement can be made by using the extended stop word set.

3 Algorithms for Query Expansion

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 proposing and implementing two query expansion algorithms.

3.1 Algorithm 1

Before we present the *Algorithm* 1, 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 Algorithm 2

We used two different sources for query expansion in *Algorithm* 2: the AcroMed database [1] and the LocusLink database [5]. Our objective of using these two databases is to find more variants of a gene name and the full name of a gene that is not available in the original TREC topics. The *Algorithm* 2 is described in Figure 2.

```
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 1 for the automatic run

```
Input: a name of gene, AcroMed and LocusLink databases
Output: a list of possible variants for the gene name

Method:
    (1) For an acronym in TREC topics, retrieve its corresponding full names from AcroMed (at most 10). Then choose 6 terms with the highest frequencies from these full names.
    (2) For these 6 terms, retrieve its corresponding acronyms from AcroMed and rank them using Okapi BM25. Then take these records which have a higher weight than a certain cut-off such as 0.8.
    (3) Put all the acronyms generated by Step 2 into a list corresponding to AcroMed
    (4) For an acronym in TREC topics, retrieve its aliases and synonyms from LocusLink database. Then put all these aliases and synonyms into a list correspond to LocusLink.
    (5) Merge these two lists genereted by Step 3 and 4 together.
    (6) Manually remove some inaccurate names from the merged list generated by Step 5.
```

Figure 2: The algorithm 2 for the manual run

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 six runs in total for the 2005 TREC Genomics track. Only the first two runs were contributed to the pools for assessing purpose. The first run is automatic and the second run is manual. The automatic run uses *Algorithm* 1 for query expansion and the manual run uses *Algorithm* 2 for query expansion.

The mean average precision (MAP) of our automatic run "york05ga1" is 0.2888 and the mean average precision of our manual run "york05gm1" is 0.3020, which are presented in Table 1. Performance comparison in terms of number of topics that achieve the best results and above median results among all the 49 topics of these two official runs are shown in Table 2.

Run	Description	Num_Rel_Retrieved	MAP	R-Prec
york05ga1	automatic	3,534	0.2888	0.3118
york05gm1	manually	3,667	0.3020	0.3212

Table 1: Official results at the 2005 Genomics track

We conduct more experiments to investigate the influence of using pseudo-relevance feedback, the influence of using an extended stop word set and the influence of using the merging technique. Detailed results are presented in Table 3. The first row describes the performance for the run without using pseudo-relevance feedback, which is the base run for comparison. The second row is our official automatic submission with pseudo-relevance feedback. The 3rd row is an improved version over the

Run	Description	Best	> Median
york05ga1	automatic	3	40
york05gm1	manually	18	48

Table 2: Performance comparison of 49 topics on the 2005 Genomics datasets

second one by using an extended stop word set and fixing a bug in our program. The extended stop word set includes 416 stop words provided Okapi plus six new stop words ("gene", "role", "impact", "biological", "disease" and "process"). The 4th row is our official manual submission and the 5th row is the merged result from the 3rd one and 4th one. The value in the parentheses is the relative rate of improvement over the base run.

Run	Description	Num_Rel_Retrieved	MAP	R-Prec
york05ga1-without	without PRF	3,469	0.2640	0.2796
york05ga1	with PRF	3,534	0.2888 (9.39%)	0.3118 (11.52%)
york05ga1-improved	improved	3,632	0.3011 (14.05%)	0.3237 (15.77%)
york05gm1	manual	3,667	0.3020 (14.39%)	0.3212 (14.88%)
york05gam-merged	merging	3,747	0.3136 (18.79%)	0.3305 (18.21%)

Table 3: More results on the 2005 Genomics datasets

5 Conclusions

The contributions of our work are as follows. First, we have designed and implemented two algorithms for query expansion. *Algorithm* 1 is very simple, easy to implement and don't need any external biomedical resource. *Algorithm* 2 needs to use AcroMed and LocusLink databases for query expansion. We find that both algorithms are powerful for improving retrieval performance in biomedical domain. Combining results from these two algorithms can produce a better result. Second, we demonstrate that pseudo-relevance feedback is effective in improving retrieval performance. Third, we show that using the extended stop word set can make a positive contribution for the retrieval performance.

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 Kai Zhang for his 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