UB at TREC-Genomics 2006: Using Passage Retrieval and Pre-retrieval Query Expansion for Genomics IR

Miguel E Ruiz
State University of New York at Buffalo
Department of Library and Information Studies

Abstract

This paper presents the results of the University at Buffalo (UB) in TREC genomics. For this task we used the SMART retrieval system and a pre retrieval expansion method that uses the ABGene and MetaMap tools. We tried two different weighting schemes one using pivoted length normalization (*Lnu.ltu*) and another using augmented tf-idf (*atn.ann*). The results show that performance of pivoted length normalization is very close to the median system that participated in the Genomics track. The augmented tf-idf performs significantly above the median system showing an improvement of 21%. This seems to indicate that a simpler weighting scheme could work better for retrieval of relevant passages.

Introduction

For this year our group participated in the Genomics track. We used a version of the SMART system [4] that has been modified in house to add support for ISO-Latin-1 and modern weighting schemes. We also collaborated with the NLM team and one of our runs was used in the fusion approach that was submitted by NLM [2].

The following sections describe the method used for processing the full text collection, query processing and expansion, as well as our results and conclusions.

Collection Processing

We used the preprocessed XML version of the full text articles that was made available by the NLM team. This XML format identified major sections, paragraphs and sentences within the full-text HTML document. Figure 1 shows the structure of the XML data. Since the TREC genomics task requires systems to return valid sections of the full-text documents, we decided to use paragraphs as the document unit in SMART. In past TREC conferences we have used several ctypes (i.e. title, abstract, and MeSH terms) to build a generalized vector space model representation of each document[3]. However, given the fact that our indexing unit is a paragraph we decided to use a single ctype this year. We kept the major name of the section tag, identifier, offset and length information as part of each paragraph. This information is used to prepare the results according to the format defined for submitting final results.

One of the issues we run into with this approach was that the size of the index file generated was larger than 2 GB causing SMART to fail during the indexing process. To avoid this problem we had to split the collection into three sub-collections. This means that the queries have to be run against each sub-collection and then the three sets of results have to be merged into a single list. Since all sub-collections had about the same number of documents and used the same waiting scheme we performed a straight forward linear combination that ranks the documents according to their retrieval score in the respective sub-collection.

```
<DOC FILE=”file-name”>
   <SECTION NUMBER=”num” NAME=”name”>
      <PARAGRAPH NUMBER=”num”>
         <TITLE ID=”id” NUMBER=”num”
               OFFSET=”offset” LENGTH=”len”>…</TITLE>
         <HEADER1 ID=”id” NUMBER=”num”
               OFFSET=”offset” LENGTH=”len”>…</HEADER1>
         <HEADER2 ID=”id” NUMBER=”num”
               OFFSET=”offset” LENGTH=”len”>…</HEADER2>
         <SENTENCE ID=”id” NUMBER=”num”
               OFFSET=”offset” LENGTH=”len”> ….</SENTENCE>
      </PARAGRAPH>
   </SECTION>
</DOC>
```

Figure 1 XML document format distributed by NLM

Query Processing

We used the expansion terms generated by the NLM team which included gene names and diseases generated using ABGene [5] and MetaMap [1]. Because our last year experiments did not show significant retrieval improvements using pseudo relevance feedback, we decided to use only pre-retrieval expansion.

Weighting Schemes

We tried two weighting schemes for documents and queries: i) pivoted length normalization (*Lnu.ltu*) and ii) augmented tf-idf (*atn.ann*).

For pivoted length normalization we used a slope value of 0.25 and the average length of documents within each sub-collection as the pivot value (pivot_a = 32.435, pivot_b=30.4005, and pivoc_c =33.4440). The augmented tf-idf used local idf values for each sub-collection. Due to the homogeneity of the sub-collections we decided not to add

a global idf value. The final list of results is generated by merging the retrieved results from each sub-collection according to their retrieval scores.
We also check that returned paragraphs do not cross the valid spans. We perform this check to detect possible errors in the XML conversion of the original documents.

Results

We submitted two official runs UBexp1 and UBexp2. The first run was produced using pivoted length normalization while the second run was generated using augmented tf-idf. Since we had not worked with passage retrieval before we were not sure which weighting scheme would perform better. Table 1 shows the performance of our official runs and two baseline runs that use the original queries (without expansion) and the performance of the median system in TREC Genomics. Our results show that pre-retrieval expansion improves results significantly (8% for the *Lnu.ltu* weighting scheme and 10% for the *atn.ann* runs). The performance of the atn.ann runs als show significant improvements over the Lnu.ltu runs and the median system (21%). Figure 2 shows a query by query comparison between the best official run (UBexp2) and the median system using the document MAP. It seems clear that this run performs well above the median system.

Our results also show that document length normalization does not seem to give any advantages. This could be due to the fact that our retrieval unit is a paragraph and their size does not vary as dramatically as full length articles. Also the augmented-tf-idf seems to be taking advantage of the fact that shorter paragraph are ranked higher than long paragraph which tend to contain more focused information.

Table 1 Mean Average Precision (official runs indicated by *)

Run	Document	Passage	Aspect
***UBexp1 (*Lnu.ltu*)**	0.2784	0.0348	0.1593
***UBexp2 (*atn.ann*)**	**0.3364**	**0.0403**	**0.1922**
UBbaseline1(*Lnu.ltu*)	0.2580	0.0269	0.1506
UBbaseline2(*atn.ann*)	0.3068	0.0323	0.1906
TREC Median system	0.2790	0.0240	0.1169

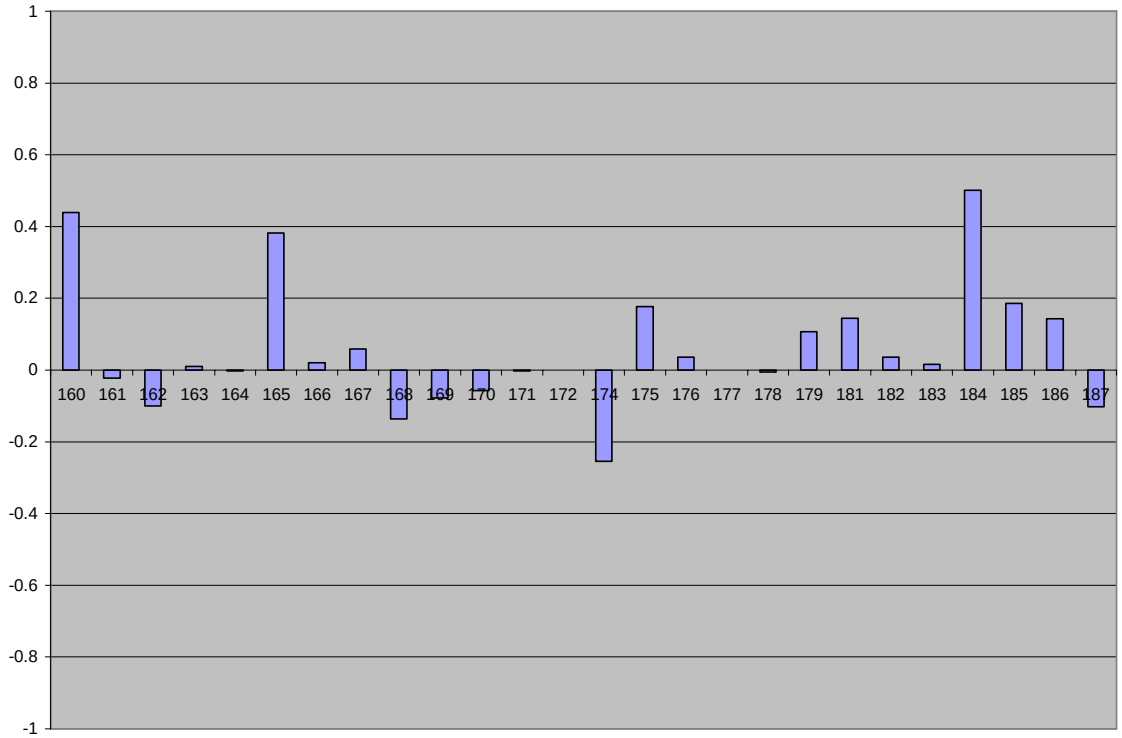

Figure 2 Difference of UBexp2 with respect to the median system

Conclusions

We presented here our results on the TREC genomics track. Although we used a relatively simple approach our results show a good level of performance for the augmented tf-idf runs. Also pre-retrieval expansion shows notable improvements. We plan to look further into the reasons why the traditional atn.ann performed well above the pivoted length normalization but so far it seems that the retrieval unit (paragraphs) does not show large variations as full length documents.

References

1. Aronson, A., Effective Mapping of Biomedical Text to the UMLS Metathesaurus: The MetaMap Program. in *American Medical Informatics Association Annual Symposium*, (2001).
2. Demner-Fushman, D., Humphrey, S.M., Ide, N.C., Loane, R.F., Ruch, P., Ruiz, M.E., Smith, L.H., Tanabe, L.K., Wilbur, W.J. and Aronson, A.R., Finding relevant passages in scientific articles: fusion of automatic approaches vs. an

interactive team effort. in *TREC 2006 conference*, (Gaitherburg, MD, 2006), NIST.

3. Ruiz, M.E. *Experiments on genomics ad-hoc retrieval*. NIST, Gaithersburg, MD, 2005.
4. Salton, G. (ed.), *The SMART Retrieval System: Experiments in Automatic Document Processing*. Prentice Hall, Englewood Cliff, NJ, 1971.
5. Tanabe, L. and Wilbur, W.J. Tagging gene and protein names in biomedical text. *Bioinformatics*, *18* (August 2002). 1124 – 1132.