NLPR at TREC 2005: HARD Experiments

Bibo Lv, Jun Zhao
National Laboratory of Pattern Recognition,
Institute of Automation, Chinese Academy of Sciences
Beijing, China 100080
{ bblv, jzhao }@nlpr.ia.ac.cn

1 Overview

It is the third time that Chinese Information Processing Group of NLPR takes part in TREC. In the past, we participated in Novelty track and Robust track, in which we had evaluated our two key notions: Window-based Retrieval Algorithm and Result Emerging Strategy [1][2]. This year we focus on investigating the significance of relevance feedback, so HARD track is our best choice.

HARD2005 is very different from that in the past two years. Firstly, Metadata is removed from topic description so that the topic description in HARD is the same as that of Robust track. Secondly, passage retrieval is cancelled this year.

The paper introduces our work on HARD Track in TREC 2005, mainly (1) we propose a new feature selection method for query expansion in relevance feedback; (2) we adopt some query expansion methods.

Our paper is organized as follows. Section 2 introduces our system, a new term selection algorithm for query expansion, and our clarification forms. Section 3 presents our query expansion methods. In section 4 experimental results are given, and finally we conclude our work in section 5.

2 System Introduction

2.1 Retrieval Model

As to the retrieval model, Lemur toolkit developed by UMASS and CMU includes six different retrieval models [3]. In order to facilitate our work, we use Okapi BM25 [4][5] as the retrieval model, which is based on the probability model of Robertson and Sparck Jones. The formula is described as follow:

$$w = \frac{(k_1+1)tf}{K+tf}\frac{(k_3+1)qtf}{k_3+qtf} + k_2 \cdot |Q| \cdot \frac{avdl-dl}{avdl+dl} \qquad (1)$$

$$K = k_1((1-b)+b \cdot dl/avdl)$$

Where,

Q is a query, containing several query terms.
w is the weight of a term in the query.
tf is the frequency of a term in a specific document.
qtf is the initial weight of the term.
avdl is the average document length.
dl s document length.
Others are parameters which depend on the query and testing data.

2.2 A New Method for Term Extraction in Relevance Feedback

As we know, pseudo relevance feedback can improve the performance of IR system. Unfortunately, much noise will also be introduced which will decrease the performance of IR system based on pseudo relevance feedback Therefore, term selection is very important in IR model. **Robertson Selection Value (RSV)** [6] is a well known selection function in probability model which can be expressed by equation 2.

$$sv(t) = (p - \overline{p}) \log(\frac{p(1-\overline{p})}{\overline{p}(1-p)}) \approx r \log(\frac{p(1-\overline{p})}{\overline{p}(1-p)}) \qquad (2)$$

$$p = \frac{r}{R}, \overline{p} = \frac{n-r}{N-R}$$

Where:

t is a term.
sv(t) is RSV of t.
r is the number of relevant documents containing the term.
R is the number of relevant documents.
n is the number of documents containing the term.
N is the number of documents within testing data.
p denotes the probability of the term in relevant documents.
$\overline{p}$ denotes the probability of the term in non-relevant documents.

In Robertson Selection Value Model, terms have the same weight if they appear in a document no matter how many times. But sometimes this is not the truth. If a term appears with higher frequency in a relevant document, this term may carry more important information than lower frequent ones. For example, I googled “Kaifulee” when he just left Microsoft. The first ten results returned from Google only contain one web page which told me Kaifulee joined Google. Obviously, the term frequency of “google” in the retrieved documents is higher than other words. But when we use the probability model for query expansion which only considering whether the term appears in the document, the term “google” can be hardly used as expanded term, which is not our expectation. To solve the problem of RSV Model, we propose a novelty query expansion algorithm based on language model [7] which term

frequency is took into account, which can be expressed in the following fomula.

$$p' = \frac{\sum_{d \in RelevantDocs} tf_d(t)}{\sum_{d \in RelevantDocs} |d|}$$

$$\overline{p}' = \frac{\sum_{d \in \overline{FeedbackDocs}} tf_d(t)}{\sum_{d \in \overline{FeedbackDocs}} |d|}$$

Where:

$tf_d(t)$ is term frequency in a specific document.

d is a document.

|d| is document length.

p' denotes the probability of the term in relevant documents.

$\overline{p}'$ denotes the probability of the term in non-relevant documents.

In formula (3), we use p' and $\overline{p}'$ to replace p and $\overline{p}$ in formula (2) respectively, and the new function is what we use in HARD track.

$$sv(t)' = r \log(\frac{p'(1-\overline{p}')}{\overline{p}'(1-p')}) \qquad (3)$$

Where:

sv(t)' is selection value of t.

r denotes term coverage in relevant documents.

the rest denotes the relevance to relevant documents.

2.3 Clarification Form

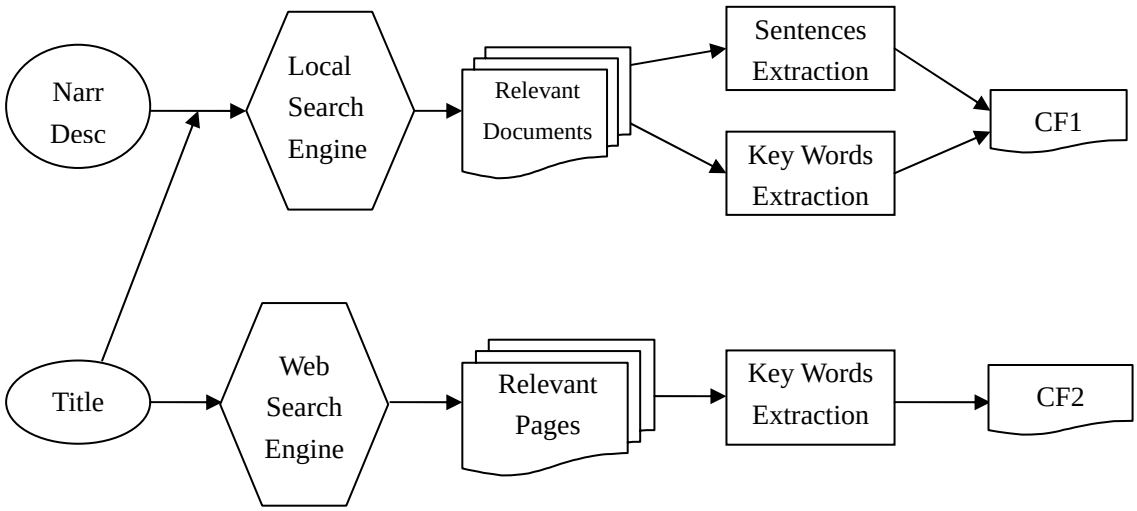

Figure 1 CF Generation

As showed in Figure 1, two sets of clarification forms were submitted. One set is used to get feedback information from relevant documents within the AQUAINT corpus, the other is set up to get feedback information returned from Google. The information submitted to assessors is mainly composed of two forms. One is key words that may be relevant to the given topic and the other is sentences extracted from the headline field of relevant documents.

3 Implementation of Query Expansion in Our Experiment

The general idea of query expansion in relevance feedback is to re-estimate query based on the distribution of terms in relevant documents and irrelevant documents, next re-estimate the query to form a new query, and last put the new query into information retrieval system to get the final results. Two expansion strategies are proposed in the following.

3.1 Using Relevant Words for Query Expansion

As told in section 2.3, after clarification forms' return-back, what we can obtain from CF1 are the sentences and the keywords that are relevant to its topic. One direct way for estimating the new query is combination of the original query and some key words judged by assessors. This is regarded as **expansion 1**.

CF2 contains information gotten from WEB, which provides us many resources. How to make full use of them to improve IR system is a challenge. Google is a powerful search engine which retrieves relevant pages in Internet according to the query. Many researchers have studied how to make use of Google to improve their own IR systems [8].

In our experiments, we only put the words in title field of topic into Google and fetch the top 20 web pages given by Google. Then 10 terms are extracted from relevant pages according to the number of returned pages containing the term. These words are extracted from WEB. After getting rid of noises by assessors and adding additional keywords, we combine original query with key words obtained from CF2 to form a new query. This is regard as **expansion 2**.

3.2 Using Relevant Sentences for Query Expansion

In CF1, there are sentences that are relevant to the given topic and extracted from field HeadLine of retrieved documents based on original query. That is to say, if a

sentence is judged as relevant, its document is relevant too.

In order to make use of the relevant documents judged by assessors for query expansion, our basic idea is to extract expanded words from these documents based on formula (3). This is regarded as **expansion 3**.

In the method of expansion 3, it is possible that r in formula (3) is not took into account for some topics, because nearly all the relevant document candidates are judged as noise by assessors. Therefore, in order to make good use of formula (3) for query expansion, new relevant documents must be found based on the given relevant ones. The method that we adopt is as follows.

First, we convert documents into vectors in the vector space model: $d_i=(t_{i1},t_{i2}...)$, next, compute the centroid of the relevant documents according to formula 4, and then score the documents in cosine similarity, the relevant documents from which the new query is built are excluded from ranking.

$$\text{centroid} = \frac{\sum_{d_i \in RelevantDocs} d_i}{R} \qquad (4)$$

$$score_i = sim(\text{centroid}, d_i) = \frac{d_i \cdot \text{centroid}}{|d_i||\text{centroid}|} \qquad (5)$$

Finally, Key words are extracted from top 10 documents according to formula (3) and this is regarded as **expansion 4**.

4 Experimental Results & Analysis

We submitted 10 runs in all. All the queries are constructed from all fields of given topics automatically. The following table describes all the submitted runs. NLPRB is our baseline which uses pseudo feedback, the others are final runs.

ID Tag	Expansion 1	Expansion 3	Expansion 4	Expansion 2
NLPRB	No	No	No	No
NLPRCF1	Yes	Yes	Yes	no
NLPRCF1CF2	No	Yes	No	yes
NLPRCF1S1	No	Yes	No	No
NLPRCF1S1CF2	Yes	No	No	Yes
NLPRCF1S2	No	No	Yes	no

NLPRCF1S2CF2	No	No	Yes	Yes
NLPRCF1W	Yes	No	No	No
NLPRCF1WCF2	Yes	No	No	Yes
NLPRCF2	No	No	No	Yes

Figure 2. Run Description

The performances of 10 Runs are shown in Figure 4. Note that, the parameters k_1, k_2, k_3, b of the model are set as 1.2, 0, 7 and 0.75 respectively.

ID Tag	R-Precision	P@10	Average Precision
NLPRB	0.2942	0.4840	0.2586
NLPRCF1	0.3336	0.5820	0.3007
NLPRCF1CF2	**0.3514**	**0.6060**	**0.3179**
NLPRCF1S1	0.3074	0.5440	0.2745
NLPRCF1S1CF2	0.3429	0.5800	0.3105
NLPRCF1S2	0.3186	0.5620	0.2818
NLPRCF1S2CF2	0.3441	0.5840	0.3088
NLPRCF1W	0.3154	0.5000	0.2631
NLPRCF1WCF2	0.3318	0.5600	0.2876
NLPRCF2	0.3234	0.5440	0.2745

Figure 3. Run Results

From the comparative results of the 10 submitted runs, we can get some conclusions from the following 4 aspects:

a) In comparison with the baseline, R-Precison, P@10 and Average Precison of NLPRCF1W Run are improved by 7.2%, 3.3% and 1.7% respectively. We believe that the reason lies in that noise removal for query expansion conducted by assessors.

b) NLPRCF2 shows that the external information can improve the system performance. But the improvement is not as well as it was expected. In fact, two approaches can be adopted to improve the performances. Firstly, maybe we should not only put the words within title into Google, because information from other fields may also contribute to better performance. Secondly, merging the retrieval results from many searching engines may be also useful, because many search engines will give you different top N pages for a specific query.

c) NLPRCF1S2 outperforms NLPRCF1S1, this validate our comparison of expansion 3 and expansion 4. However, there is an interesting phenomenon that

when they are combined with CF2, the results are hard to explain. We do not know clearly what it happens.

d) NLPRCF1CF2 outperforms others in all the 10 runs which combine the information obtained from relevant documents and retrieval results from Google. It gets increases of 19.4% for R-Precision, 25.2% for P@10 and 22.9% for Average Precision. It is obviously that the more information we use the better performance we will obtain.

Because the best, median and worst results of all 50 topics are not given, we choose first 10 topics to compare.

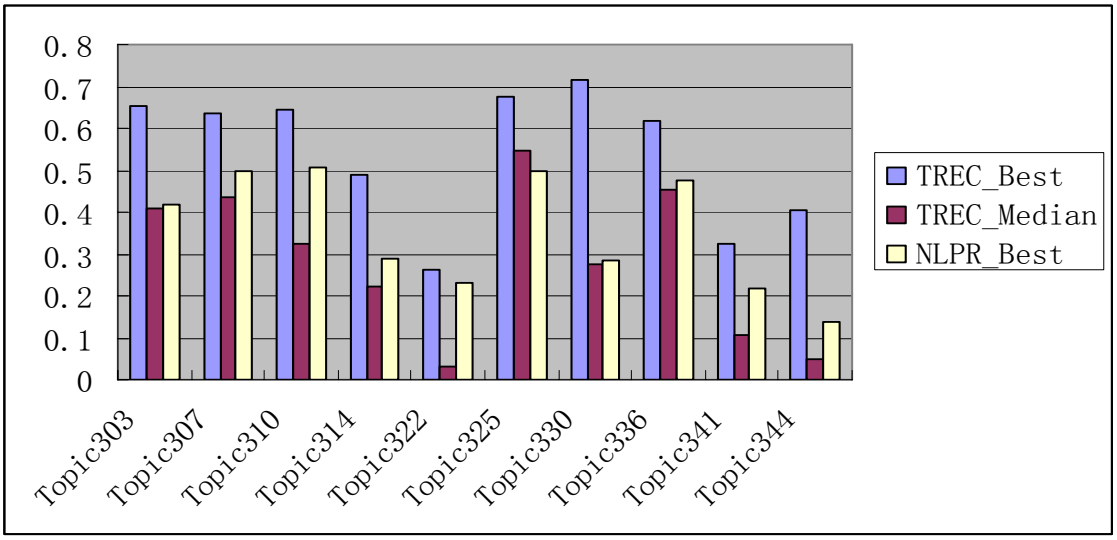

Figure 4. R-Precision Comparison

Figure 4 tells us that our result is only above median and there is a lot of work to do in the future. We should pay special attention to the following three key points from comparison:

a) Query expansion should be in the form of not only words but also phrases such as baseNP, which often improves the performance of system significantly.

b) New feature selection function which removes noise in relevant documents should be studied.

c) Further research will be focused on mining web resources for information retrieval.

4. Conclusions

We propose a new feature select function and evaluate two kinds of query expansion methods, all of which can improve our system performance to some extent via relevance feedback. The experiments show that all these techniques are effective, especially combination of query expansion based on relevant information from web

and testing data.

Future work will focus on query expansion based on phrase and more effective feature selection methods.

5. Acknowledge

This work was supported by the Natural Sciences Foundation of China under grant No. 60372016, the Natural Science Foundation of Beijing under grant No. 4052027.

6. Reference

[1] Qianli Jin, Jun Zhao, Bo Xu. NLPR at TREC 2003 - Novelty and Robust Track. Text Retrieval Conference (TREC-2003), NIST, Maryland, USA, 2003.

[2] J. Xu, J. Zhao, B. Xu, Chinese Academy of Science NLPR at TREC 2004: Robust Experiments. Text Retrieval Conference (TREC-2004), NIST, Maryland, USA, 2004.

[3] Http://www.lemurproject.org

[4] S E Robertson, S Walker, M Beaulieu. Okapi at TREC-7: Automatic Ad Hoc, Filtering, VLC and Interactive. Text Retrieval Conference (TREC-7), NIST, Maryland, USA, 1998.

[5] E Robertson and S Walker. Okapi/Keenbow at TREC-8. The Eighth Text REtrieval Conference (TREC-8), NIST, Maryland, USA, 1999.

[6] S E Robertson. On Term Selection for Query Expansion. Journal of Documentation, 46:359-364, 1990.

[7] J.M.Ponte and W.B.Croft, A Language Modeling Approach to IR, In the Proceedings of the 12^{th} ACM SIGIR Conference, pp.275-281,1998.

[8] K.L. Kwok, L. Grunfeld, H.L. Sun and P. Deng TREC2004 Robust Track Experiments Using PIRCS. Text Retrieval Conference (TREC-2004), NIST, Maryland, USA, 2004.