The Hong Kong Polytechnic University at the TREC 2004 Robust Track

D.Y. Wang, R.W.P. Luk, K.F. Wong[1]

Department of Computing
The Hong Kong Polytechnic University

[1]Department of Systems Engineering and Engineering Management
The Chinese University of Hong Kong

ABSTRACT

In the robust track, we mainly tested our passage-based retrieval model with different passage sizes and weighting schemes. In our approach, we used two retrieval models, namely the 2-Poisson model using BM25 term weights and the vector space model (VSM) using adaptive pivoted unique document length normalization. Also, we utilize WordNet to re-weight some PRF terms and extract some context words as expanded query terms. We show that our passage-based model achieves the comparable performance on the whole query set. Moreover, our new methods of using WordNet information for query expansion can improve the retrieval performance.

1. INTRODUCTION

Generally speaking, there are two major categories of information retrieval technology and research: semantic and statistical.

Semantic approaches mainly depend on syntactic and semantic information and they try to understand the natural language text that a user would provide. In statistical approaches, the retrieval is based on some statistical measure of query, documents and relation between them. By far most work to date has been devoted to statistical approaches. It is well acknowledged that statistical approaches combined with some semantic methods will yield better performance.

In the past, lots of people use WordNet for information retrieval. [Voorhees 93] used synsets in WordNet to disambiguate nouns in documents and then used a combination of synonyms, hypernyms and hyponyms for query expansion [Voorhees 94]. [Stairmand 97] used WordNet to compute the lexical cohesion for IR. [Richardson 95] used WordNet to calculate the semantic distance between concepts or words in order to get the similarity between a documents and a query. [Mandala 98] investigate the problem of low performance improvement by using WordNet and tried to use automatically constructed thesaurus to compensate the insufficiency of

information of WordNet. [Flank 98] proposed a layered approach for IR and used WordNet for semantic expansion.

[Liu, et al 04] utilized WordNet to disambiguate word senses of query terms and after the sense is determined, they use WordNet to expand the query. They use definition words as well as related words that others have often used, such as synonyms.

In addition to our passage-based model, we also make use of WordNet. In detail, we use the definition as well as all relationships defined for each of the senses: noun, verb, adjective and adverb in WordNet. Section 3 and 4 will explain these implementations.

2. FORMAL RUNS AND OTHER RUNS

Our formal runs were based on passage retrieval. Each passage has fixed length of 300 terms, unless we encountered the end of file before. The document similarity score *sim(.)* is computed by combining passage scores using a weighted Boolean disjunction operation [Fox 92] or generalized mean function:

$$sim(d_i, q) = \sqrt[\alpha]{\sum_{j=1}^{k_i} rel(p_{i,j}, q)^{\alpha}}$$

where q is the query, d_i is the i-th document, $p_{i,j}$ is the j-th passage of the i-th document, k_i is the number of passages in the i-th document, *rel(.)* is the relevance score assigned by the retrieval model, and α (= 20) is a soft-hard decision parameter. We used two retrieval models to return *rel(.)*, namely the 2-Poisson model using BM25 term weights [Robertson et al., 1996] and the vector space model (VSM) using adaptive pivoted unique document length normalization [Singhal 96]. In addition, the retrieval was carried out with pseudo-relevance feedback (PRF) as in our Chinese retrieval [Luk 04] where the top 20 passages were analyzed, selecting 40 terms and using a term weight mixture parameter of 0.3.

Run No	Run Name	Model	Query	Submitted	
				MAP	P@10
1	polyutp1	BM25	T	0.2215	0.3916
3	polyutp3	VSM	T	0.2308	0.4137
2	polyudp2	BM25	D	0.1948	0.3671
4	polyudp4	VSM	D	0.1945	0.3791
5	polyudp5	BM25	TDN	0.2455	0.4422
6	polyudp6	VSM	TDN	0.2383	0.4418

Table1: Performance of our six submitted runs.

The formal runs were obtained when we had some problems with the stop word lists and the passage size was adjusted to 250 terms per passage. We re-ran the same

queries for some of the runs and Table 2 showed that better results were obtained. More significant improvement was found to be in the description queries. The performance of our basic retrieval system was performing similar to the median performance in this robust track compared with the formal runs only (see Table 3).

Run No	Run Name	Model	Query	Informal runs	
				MAP	P@10
1’	Polyutp1	BM25	T	0.2566	0.4365
2’	Polyudp2	BM25	D	0.2495	0.4390
5’	Polyudp5	BM25	TDN	0.2739	0.4924

(note: passage size =250, PRF)

Table2: Performance of our informal runs.

Table 3 compares the retrieval effectiveness performances between our search engine and the search engine by other participants. Our search engine was approximately in the middle (close to the median) for the MAP and for the P@10 performances. The differences between the best formal runs by other participants were statistically better than ours.

Run No	Query Type	MAP (%)					P@10 (%)				
		Our Runs		All Participants’ Runs			Our Runs		All Participants’ Runs		
		F	O	B	M	W	F	O	B	M	W
1’	T	22.2	25.7	33.3	25.4	8.0	39.2	43.7	51.3	43.6	15.4
3	T	23.1	N/A				41.4	N/A			
2’	D	19.5	25.0	33.4	26.9	7.6	36.7	43.9	51.5	45.5	18.2
4	D	19.5	N/A				37.9	N/A			
5’	TDN	24.6	27.4	35.9	27.6	7.6	44.2	49.2	54.1	45.1	15.4
6	TDN	23.8	N/A				44.2	N/A			

(Key: F for Formal, O for Others, B for Best, M for Median, W for Worst)

Table3: Comparison of MAP with others

It is possible that our search engine performed particularly poorly for a specific set of queries. Therefore, we compared the performance of our search engine with the performance of other participants’ for the three subset of queries, namely the new topics (Table 4), the old topics (Table 5) and the hard topics (Table 6). For each of the subset of queries, the performance of our search engine was close to the median. Therefore, our search engine did not perform poorly in any of the three subsets of queries.

Run	Query	Measure	New	Best	Median	Worst

No			Topic			
1’	T	MAP	0.2714	0.4019	0.2856	0.0529
		P@10	0.4102	0.5490	0.4245	0.1143
2’	D	MAP	0.2964	0.4074	0.2992	0.1015
		P@10	0.4653	0.5510	0.4633	0.2143
5’	TDN	MAP	0.3093	0.4227	0.2979	0.0529
		P@10	0.4816	0.5510	0.4449	0.1143

Table 4: Comparison of our other runs on new topics with other participants

Run No	Query	Measure	Old Topic	Best	Median	Worst
1’	T	MAP	0.2529	0.3165	0.2468	0.0865
		P@10	0.4430	0.5050	0.4370	0.1635
2’	D	MAP	0.2380	0.3158	0.2634	0.0692
		P@10	0.4325	0.5080	0.4535	0.1745
5’	TDN	MAP	0.2653	0.3429	0.2667	0.0692
		P@10	0.4950	0.5395	0.4510	0.1635

Table 5: Comparison of our other runs on old topics with other participants

Run No	Query	Measure	Hard Topic	Best	Median	Worst
1’	T	MAP	0.1187	0.1942	0.1152	0.0346
		P@10	0.2780	0.3760	0.2800	0.1040
2’	D	MAP	0.1127	0.1635	0.1328	0.0207
		P@10	0.2820	0.3820	0.3160	0.0940
5’	TDN	MAP	0.1425	0.1949	0.1260	0.0207
		P@10	0.3480	0.4020	0.2940	0.0940

Table 6: Comparison of our other runs on hard topics with other participants

3. PRF TERMS RE-RANKING

Pseudo-relevance feedback improves the performance of retrieval. However, not all PRF terms are so helpful since they are blindly extracted from top ranked documents in returned list. Some PRF terms can even bring down the performance. So we propose some methods to filter out the bad PRF term and add weight of good ones. These methods are based on that such an assumption: title query terms can reflect the user need and together with the word related, they can indicate user need much better.

[Mandala 98] found that polysemous words degrade the precision of information retrieval since all senses of the original query term are considered for expansion. Hence they consider to add the terms that are most similar to the entirety of query terms rather than a single query term. We also do the similar work: assign a higher

score for a candidate PRF term if it can match more sets of related terms of query terms.

3.1Re-rank PRF terms with WordNet Definition of query terms

We use dictionary definition of title query terms as a baseline to evaluate PRF terms. The definition of a term usually contains a hypernym of it and some limiting components. For example, in WordNet, "telescope" has a definition of "a magnifier of images of distant objects" and "Hubble" has "United States astronomer who discovered that (as the universe expands) the speed with which nebulae recede increases with their distance (1889-1953)".

Then, we know "telescope" is a kind of image magnifier and relates with distant objects and "Hubble" is an astronomer and related with "discover", "nebulae". The advantage of obtaining related words from definition is the additional words are quite direct and accurate, although the number of them is usually limited. And when some term has some completely distinct sense, such as "bank", the methods will bring in lots of noise if no word sense disambiguation techniques are taken.

Let def_1, def_2,…, def_Q be the definition of the title query q_1, q_2,…, q_Q. For a PRF term weighted prf_i, if it can match n definitions ("match" means it appears in the definition), we re-weight prf_i a new value of

$$idf_i \cdot (1 + \frac{n + 0.1}{Q + 0.1}) \cdot prf_i$$

where idf_i is idf of this PRF term.

In practice, we re-weight 60 PRF terms (i.e. Q=60). For each title query term, we get the definitions of all senses it has from WordNet. Then, we remain the nouns and stem them with Porter's Stemmer. After matching PRF terms, we rank them by new PRF weights.

We only rely on the nouns in definition because some researchers indicated that nouns are enough to represent a topic. [Brezeale 1999] believed that using nouns only will be sufficient to represent a web page. [Fell90] found that meanings of verbs are more flexible than the meanings of nouns, so the meaning of a verb is much more dependent on the kinds of nouns in the sentence.

3.2 Re-rank PRF terms with related words of query terms

The number of definition words mentioned above is small and we can only match few PRF terms to them and the effectiveness of re-weighting is not significant.
In consideration of this, we use WordNet defined relationships to get more related words of query terms. Take "telescope" for an example again, we get eight related terms. One hyponym of it is "astronomical telescope". Since we get "astronomer" from definition of "Hubble", we can know one of the common concept for "telescope" and "Hubble" might be "astronomy", which could be obtained by stemming both and matching.

Let $s_1, s_2, \ldots, s_Q$ be the sets of related words of the title query definitions $def_1, def_2, \ldots, def_Q$. For a PRF term weighted prfi, if it is the element of n sets among $s_1, s_2, \ldots, s_Q$, we re-weight prfi a new value of

$$idf_i \cdot (1 + \frac{n + 0.1}{Q + 0.1}) \cdot prf_i$$

where idfi is idf of this PRF term.

Let wi is a word with POS_i in Wordnet and n relationships $R_1, R_2, \ldots, R_n$ are defined for w_i. We use $R_j(w_i)$ to represent the word(s) that have relationship R_j with w_i. So for a set of word S, we get its directly related words:

$$REL_1(S) = \bigcup_{w_i \in S} \bigcup_j R_j(w_i)$$

For REL_k, we can get its directly related words set REL_{k+1} by

$$REL_{k+1}(S) = \bigcup_{w_i \in REL_k(S)} \bigcup_j R_j(w_i)$$

The above set s_i can be equal to $REL_k(def_i)$ given the iteration times k.

In practice, we use the 60 definitions generated in last section and perform POS tagging on them by MontyLingua. We get the related words of all nouns, verbs, adjectives and adverbs in definition by WordNet. These related words are generated by all relationships defined as REL_k above. Actually, our first experiments use REL_1(def) as related words. Similarly, then, we select the nouns among them and stem them for matching.

4. QUERY EXPANSION WITH NOUN PHRASES IN CONTEXT

No matter how many topics a document covers, in a small passage of it, we assume it covers only one topic. And around the position of query terms appears, the topic is more likely to be relevant with the information need. That's the reason that we extract context words around each query terms in documents. We use such context word as a supplementary for query terms just as PRF terms do. And we select some candidates from them and give them weights based on some syntactic and semantic information.

We extract context from top ranked documents in retrieved list for they are more likely to be relevant and contain more query terms. First, these documents are POS tagged and then labeled with phrases by MontyLingua. As for each query term, we sequentially extract ten non-stopword terms in both the left and right side of it. Then, we remain the terms that appears in noun phrases as the candidate terms for later matching.

4.1 Matching context terms with WordNet Definition of query terms

For context terms, we use the similar weighting methods just as we process the PRF terms. Let $def_1, def_2, \ldots, def_Q$ be the definition of the title query $q_1, q_2, \ldots, q_Q$. For a context term, if it can be found in n definitions, we give it a weight of

$$idf_i \cdot (1 + \frac{n + 0.1}{Q + 0.1})$$

where idfi is idf of this context term.

For easy comparsion, we assign weights for 60 context terms. After that, we rank them by new weights.

4.2 Matching context terms with related words of query terms

For the same reason, we weight context terms based on the results of matching them with the related words of query terms.

And we generate the related words of definitions of query terms def_1, def_2,…, def_Q: $REL_1(def_1)$, $REL_1(def_2)$, …, $REL_1(def_Q)$ and give context words weights based on the number of matching n:

$$idf_i \cdot (1 + \frac{n + 0.1}{Q + 0.1})$$

where idf_i is idf of this context term.

5. EXPERIMENTAL RESULTS

We produce 60 PRF terms from query terms by the methods used in [Luk 04]:

$$S_3(j) = \begin{cases} S_1(j) & for \quad n_j \geq k_1 \\ 2S_1(j) & for \quad 1 < n_j < k_1 \\ 0 & for \quad n_j = 1 \end{cases}$$

where $S_1(j)$= tf_j in top N doc * df_j in the topN doc* idf_j and n_j is the j-th document frequency and k_1 is a parameter of term specificity.

Then we match these PRF terms with the definition and related terms mentioned above and re-rank them. We choose the top 30 ranked PRF terms and all 60 terms to do retrieval on 10 queries (query No 311-320). The original PRF terms and their weights are set to be the baseline.

		Re-weighted by definition		Re-weighted by related terms	
	Baseline	Top30	Top 60	Top 30	Top60
MAP	0.2068	0.1617	0.2189	0.1559	0.2161
P@10	0.3600	0.3300	0.4300	0.3100	0.4300

Table 9: Performance with different sizes of terms.

We found that our re-weighting methods improve the retrieval performance a little. And we notice that retrieval with 60 terms performs much better than 30 terms. This shows that the matching methods are effective to add weights for good term but insufficient to penalize and eliminate the bad terms. Later, we found the same and

even more significant phenomenon on matching context terms. It results from the very low matching percentage.

We extend the 10 queries to 50 queries (No 301-350) of TREC-6 to show the performance of our re-weighting methods.

	Baseline	Re-weighted by definition	Re-weighted by related terms
MAP	0.1301	0.1405	0.1400
P@10	0.3020	0.3200	0.3200

Table 10 : Peformance on TREC-6

In order to investigate the matching of context words, we extracted 10349 terms from noun phrases in the context of query terms (No 317-333) and each query has 608 distinct context terms in average. In total, only 160 of the 10349 context terms can match the definition terms so the matching percentage is 1.5%. And 650 of the 10349 context terms can match related terms (REL_1) of the query terms so the matching percentage is 6.28%.

6. CONCLUSION

From above test and evaluation of our formal runs, other runs, we find the our formal runs perform badly because of the stopword list failure. The performance of our other runs is comparable with the median of all participants on T and TDN queries and ours is a little worse than median with D queries.

Our WordNet-based experiments show that using WordNet to re-weight original PRF term can improve the results but it still needs more work if we expect significant progress. The low percentage of matching context words requires other term expansion methods to make it reasonable and feasible.

ACKNOWLEDGEMENT

We thank Prof. Kwok for his participation in the project PolyU 5199/04E which supported this work. And we thank Ms WONG Wing-sze for her kind help in setting up the system.

REFERENCE

[Fell90] Fellbaum, Christiane, “English verbs as a semantic net,” in International Journal of Lexicography 3 (4):278 - 301, 1990.

[Flank 98] Sharon Flank. A layered approach to NLP-based information retrieval. Proceedings of the 17th international conference on Computational linguistics - Volume 1 Montreal, Quebec, Canada 1998

[Fox 92] E. Fox, S. Betrabet, and M. Koushik, "Extended Boolean models ", In W. Frakes and R. Baeza-Yates, Information Retrieval: Data Structures and Algorithms, NJ: PHI, pp. 393-418, 1992

[Liu, et al 04] Shuang Liu, Fang Liu, Clement Yu, Weiyi Meng. An effective approach to document retrieval via utilizing WordNet and recognizing phrases. Proceedings of the 27th annual international conference on Research and development in information retrieval. July 2004.

[Luk 04] Robert W. P. Luk and K.F. Wong. Pseudo-Relevance Feedback and Title Re-ranking for Chinese Information Retrieval. Working Notes of NTCIR-4, Tokyo, 2-4 June 2004.

[Mandala 98] Rila Mandala, Takenobu Tokunaga, and Hozumi Tanaka. The use of WordNet in information retrieval. In Sanda Harabagiu, editor, Use of WordNet in Natural Language Processing Systems: Proceedings of the Conference, pages 31--37. Association for Computational Linguistics, Somerset, New Jersey, 1998.

[Richardson 95] R. Richardson and A.F. Smeaton. 1995. Using wordnet in a knowledge-based approach to information retrieval. Technical Report CA-0395, School of Computer Applications, Dublin City University.

[Robertson et al., 1996] Robertson, S.E., Walker, S., Beaulieu, M.M., Gatford, M., Payne, A. Okapi at TREC-4, *The Fourth Text REtrieval Conference (TREC-4)*, NIST Special Publication 500-236, National Institute of Standards and Technology, Gaithersburg, MD, pp. 73-86, October 1996.

[Scott98] Scott, Sam and Stan Matwin, "Text classification using WordNet hypernyms," Proceedings of the COLING/ACL Workshop on Usage ofWordNet in Natural Language Processing Systems, Montreal, 1998.

[Singhal 96] Singhal, A., Buckley, C., Mandar, M. Pivoted Document Language Normalization, *Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval,* pp.21-29, 1996.

[Stairmand] M.A. Stairmand. Textual context analysis for information retrieval. In Proceedings of the 20th A CM-SIGIR Conference, pages 140--147. 1997.

[Voorhees 93] E.M. Voorhees. Using wordnet to disarmbiguate word senses for text retrieval. In Proceedings of the 16th A CM-SIGIR Conference, pages 171-180. 1993.

[Voorhees 94] E.M. Voorhees. Query expansion using lexical-semantic relations. In Proceedings of the 17th ACM-SIGIR Conference, pages 61-69. 1994