Experiments with Query Expansion at TREC 2006 Legal Track

Feng Charlie Zhao, Yugyung Lee, Deep Medhi
School of Computing and Engineering
University of Missouri - Kansas City
fczrzf,leeyu,dmedhi@umkc.edu

***Abstract*— This paper describes the UMKC TREC 2006 Legal Track experiments. We focus on a single technique that uses co-occurrence based thesaurus to expand queries. Our results indicate this technique is effective even towards the enormous vocabulary size in the IIT CDIP collection.**

I. Introduction

The vocabulary problem [1] in document retrieval is a primary hindrance function to its performance where polysemy may reduce precision and synonymy may reduce recall. Query expansion seeks to mitigate this problem by expanding queries with additional conceptual relevant terms.

Query expansion is often associated with debatable results [2] [3] when expanding from manually compiled thesaurus like WordNet [4]. When thesaurus has been re-defined as ”any data structure that defines semantic relatedness between words” [5], people noticed some important limitations in this approach [6]. Encouraged by some positive results in this research topic [5] [7], we believe an effective query expansion model has to resolve two fundamental issues. To find those conceptual relevant terms with great affinity to original query terms and further to assign proper weights to them to reflect the overall query emphasis.

Legal Track provides a unique opportunity to evaluate such query expansion model based on co-occurrence thesaurus. The IIT CDIP collection [8] has more than 200 million unique terms in our index. It becomes questionable for any query expansion model to excel with the presence of such phenomenal noise in OCR data. Therefore we designed our experiments to explore the effectiveness and robustness of our current query expansion model.

II. Our Query Expansion Model

A. Architecture for Query Expansion

The knowledge of conceptual relevance between any two terms is a precondition for our model, but this model is applicable regardless the underlying method which yields that knowledge. In this paper, the focus is to discuss our query expansion model rather the method of computing with conceptual relevance. There is no consensus of the definition of relevance regardless of it is actually a core concept in information retrieval. There are many different school of thoughts to compute the conceptual relevance or similarity between two terms whereas our approach is primarily based on statistical analysis of co-occurrence patterns.

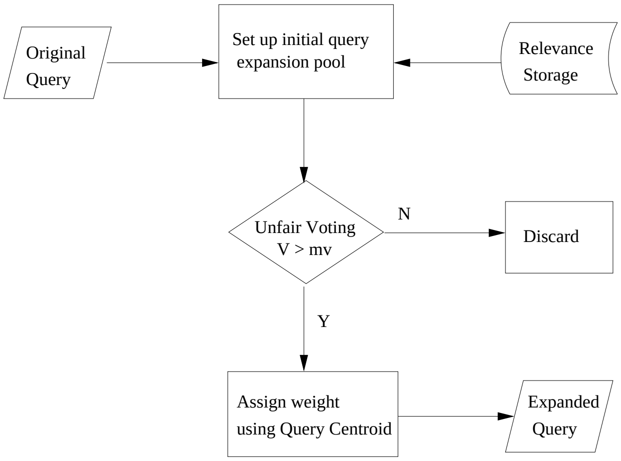

Fig. 1. Query Expansion Model

In our model, we first retrieve conceptual relevance information from the component of relevance storage according to the original query terms. That sets up the initial pool of query expansion candidates. Unfair voting component will discard all those candidates whose votes don't exceed the majority vote. Those winners will then be assigned a weight based on its correlation to the query centroid which is the unique representation of the overall emphasis of query.

B. Unfair Voting

It is so called unfair voting because the votes casted from original query terms to query expansion candidate terms carry different weights. As the result, even every vote still counts, but those terms which have multilateral conceptual relevance to leading query terms will get high votes. And they will move to the next stage to receive their query term weights.

The purpose of our unfair voting strategy is for word sense disambiguation. By marginalizing those terms which do not have multilateral relationship with the majority of original query terms, we are able to discern the correct sense based on query context. Then we can discard those candidates if they are misaligned with the current word sense.

The procedure of unfair voting has three major steps with the following definitions:

- m: number of original query terms
- MV: majority vote threshold
- q_i: weight for original query term i
- β, δ: experimental regulating parameters
- v_i: original query term i as voter
- c_j: candidate term j
- V_j: all votes for candidate term j
- $CR(v_i, c_j)$: conceptual relevance measure of two terms v_i and c_j
- CS: the set of final query expansion candidate terms

1) Determine the Majority Vote (MV) from weight (q_i) of original query terms

$$MV = \sum_i^m \alpha q_i \ , \quad \alpha = \begin{cases} 1 & q_i > \beta \frac{\sum_i^m q_i}{m} \\ 0 & \text{else} \end{cases} \tag{1}$$

2) Cast votes (V_j) for candidate term j (c_j) from all voters (v_i)

$$V_j = \sum_i^m \gamma q_i \ , \quad \gamma = \begin{cases} 1 & CR(v_i, c_j) > \delta \\ 0 & else \end{cases} \tag{2}$$

3) Form the final query expansion candidate term set (CS)

$$CS = [\cdots c_j \cdots] \ \forall \ V_j > MV \tag{3}$$

C. Query Centroid

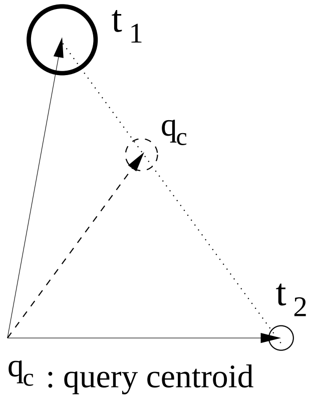

Fig. 2. Query Centroid

Query centroid is a unique overall representation of the query in our model. As shown in figure 2, it is the center of the mass where circles on the tips of vectors representing the size of mass. So the query centroid always leans to the dominant query terms. Query centroid can be obtained from equation 4 with the following definitions:

- $\overrightarrow{q_c}$: query centroid
- $\overrightarrow{t_i}$: query component i
- m_i : mass of component i
- $m_{tot} = \sum m_i$: total mass

$$\overrightarrow{q_c} = \frac{\sum \overrightarrow{t_i} m_i}{m_{tot}} \tag{4}$$

After obtaining this query centroid vector, we assign the weight to query expansion candidate terms using equation 5 with the following definitions:

- $\overrightarrow{W}$: the weight vector for all query expansion candidate terms
- C : the correlation matrix of query expansion candidate terms
- C_{ij} : the relevance measure between candidate term i to original query term j
- $\overrightarrow{q_c}^T$: the transpose of query centroid vector

$$\overrightarrow{W} = C \overrightarrow{q_c}^T \tag{5}$$

We sort the weight vector of $\overrightarrow{W}$ in descending order to obtain the final expanded query with ranked terms. This query can be fairly long, so in order to keep the query turnaround time within expectation, we chop the query to a fixed length. It becomes another tuning parameter to determine the optimal query length in different corpus.

III. Experiments and Results

A. Runs

All our runs are automatic and queries are distilled from the FinalQuery element within BooleanQuery. Queries were run again the ot element in corpus only, so there is no usage of metadata elements. Therefore all runs fall into the second category of Legal Track runs, namely Good-OCR subset, no use of metadata, automatic queries. The underlying text search engine is an extension of Lucene library [9].

We submitted eight runs to TREC while kept many others for additional benchmarking. The semantics of various runs are explained in table 1, 2 and 3.

TABLE I

Runs without Query Expansion

Runs	Conditions
Orig	Boolean query with only OR considered
UMKCB	Boolean query without proximity
UMKCSN	Boolean query without positional order for proximity
UMKCSW	Boolean query with positional order for proximity

TABLE II

RUNS WITH QUERY EXPANSION ON INDIVIDUAL TERM

Runs	**Conditions**
UMKCBQE5	query length ≤ 5
UMKCBQE10	query length ≤ 10
UMKCQE25	query length ≤ 25

TABLE III

RUNS WITH QUERY EXPANSION ON OUR MODEL

Runs	**Conditions**
QE5	query length ≤ 5
QE10	query length ≤ 10
QE20	query length ≤ 20

B. Results

The results of above runs are presented in figure 3, 4 and 5.

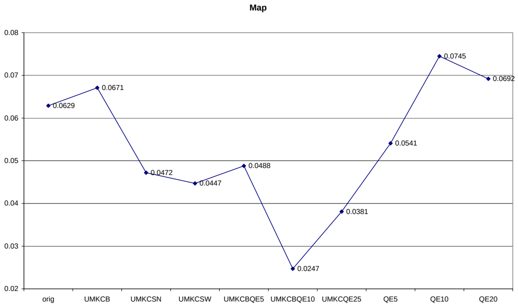

Fig. 3. Map

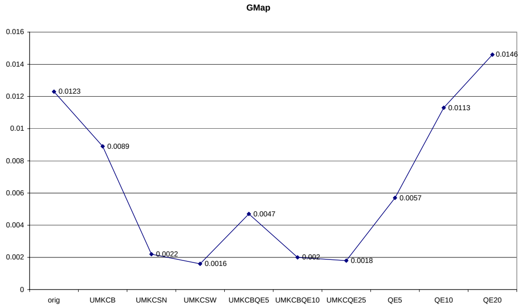

Fig. 4. GMap

C. Analysis

The experimental results indicates the boolean queries with the consideration of proximity (UMKCSN and UMKCSW) actually degraded the performance when compared with their baseline of UMKCB run. It reveals the gap between mentality of those term proximity in original boolean query to the reality of term distribution in this collection.

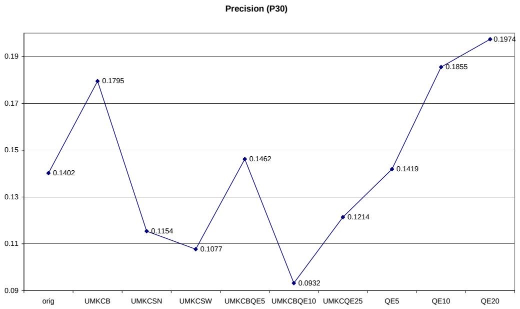

Fig. 5. Precision (P30)

The expanded queries from individual query terms can further degrade the performance by introducing in some irrelevant terms which is hard to avoid without proper means of word sense disambiguation when expanded from a co-occurrence based thesaurus.

To justify our query expansion model, we compare the performance between the expanded queries from our model to the original run as their baseline. Where we found some general improvements as shown in table 4.

TABLE IV

IMPROVEMENT FROM QUERY EXPANSION

Measures	**Most Improved**
Map	18%
Gmap	19%
P30	40%

When we investigated this improvement on individual topic, we were amazed on its effectiveness on some topics. The figure 6 of the map on topic 22 shows the technique of query expansion based on our model has improved map by almost 9 times. If we consider the statistics on this particular topic from all runs submitted in this year, then this single technique of query expansion even has better performance than topic 22's best map at 0.0303. It also demonstrated that our model is a close-fitting to this type of query from the trend that query performs better with more expanded terms.

In general, our model does not succeed in every query and in some cases, the expanded query diluted the original query intension, then degradation occurred. There are two possible reasons. First, due to the restrictions of this collection we were not able to zoom into a smaller window size to look up the co-occurrence patterns when building up the thesaurus. Therefore we had only a coarse-grained initial query expansion pool to start with. Second, our model currently missing the feature to expand the query according to its internal boolean query struc-

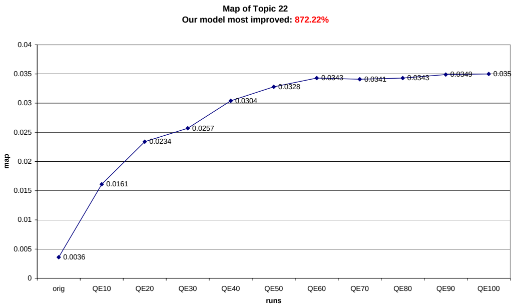

Fig. 6. Map of Topic 22

ture which we will address further in the next section.

While the results indicate our query expansion model still produce the best run within all experiments we have done, we also got disturbed by the overall poor performance of all runs in general. This means the IIT CDIP collection remains a great challenge to our query expansion model.

IV. Limitations and Future Work

Since our primary goal for this experiment is to verify the effectiveness and robustness of our query expansion model, therefore we are not expecting this one technique will be very competitive to other mature search engines which have incorporated many useful techniques. But our experiments still revealed a serious limitation in our model. Our model simply takes the query as a bag of terms and lacks the consideration of normal boolean operators of AND, OR and parenthesis which seams to be necessary to achieve better performance for complex queries in Legal Track. Therefore a specialized query parser is needed in our model to better interpret and expand legal queries.

Relevance feedback method [10] has become a standard tool for practitioners in information retrieval, but its effectiveness is largely depending on the very first run. That's why we presented precision at 30 documents (P30) in figure 5. Hypothetically these significant improvements of precision from our model should naturally become the compound interest for relevance feedback method afterwards. We are very interested to verify this hypothesis in the near future.

V. Conclusion

Our experiments in this Legal Track was designed to find out the effectiveness and robustness of our query expansion model which was powered by a novel unfair voting strategy and weighting schema based on query centroid. The results indicated even with the possession of enormous noise in the IIT CDIP data, our model still improved query performance from the baseline. With the 100% recall as one of goals in this Legal Track, we consider it is recommendable to further explore the potential of our model, especially in the direction of synthesizing with traditional relevance feedback method.

VI. Acknowledgements

The first author was partially supported by UMKC's Preparing Future Faculty Fellowship program during this research work. And we also would like to acknowledge Mr. Jim Grant and Mr. Brian Koenigsdorf from Department of Central Systems of UMKC for their tremendous support in computing resources.

References

[1] G. W. Furnas, T. K. Landauer, L. M. Gomez, and S. T. Dumais, ”The vocabulary problem in human-system communication,” Commun. ACM, vol. 30, pp. 964-971, 1987.

[2] E. M. Voorhees and Y.-W. Hou, ”Vector Expansion in a Large Collection,” in TREC, Gaithersburg, Maryland, 1992, pp. 343-352.

[3] E. M. Voorhees, ”On Expanding Query Vectors with Lexically Related Words,” in TREC, Gaithersburg, Maryland, 1993, pp. 223-232.

[4] C. Fellbaum, WordNet: an electronic lexical database. Cambridge, Mass: MIT Press, 1998.

[5] H. a. J. P. Schutze, ”A Cooccurance-based Thesaurus and Two Applications to Information Retrieval,” Information Processing and Management, vol. 33, pp. 307-318, 1997.

[6] H. J. Peat and P. Willett, ”The limitations of term co-occurrence data for query expansion in document retrieval systems,” JASIS, vol. 42, pp. 378-383, 1991.

[7] Y. Qiu and H.-P. Frei, ”Concept based query expansion,” in Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval Pittsburgh, Pennsylvania, United States: ACM Press, 1993.

[8] D. Lewis, G. Agam, S. Argamon, O. Frieder, D. Grossman, and J. Heard, ”Building a test collection for complex document information processing,” in Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval Seattle, Washington, USA: ACM Press, 2006.

[9] O. Gospodnetic and E. Hatcher, Lucene in action. Greenwich, CT: Manning Publications, 2005.

[10] J. Rocchio, ”Relevance Feedback in Information Retrieval,” in The SMART retrieval system; experiments in automatic document processing, G. Salton, Ed. Englewood Cliffs, N.J.: Prentice-Hall, 1971, pp. Chapter 14, pages 313-323.