Ad Hoc Experiments Using EUREKA

Allan Lu, Maen Ayoub, Jianhua Dong
Lexis-Nexis
A member of the Reed Elsevier plc group
9443 Springboro Pike
Miamisburg, OH 45342
(allan.lu, maen.ayoub)@lexis-nexis.com

1.0 Introduction

Our research for TREC5 focused on search and retrieval of full-text documents with short natural language (NL) queries. It has been our strong belief that the queries submitted to any operational retrieval system, especially those on the Internet, are short or very short, and that an effective approach to processing short NL queries has great application potential. We also looked at data fusion [1] with the assumption that a number of well-developed and specialized retrieval functions would probably outperform a single well-developed but general function. For example, two functions, one specialized in retrieving medium to long documents and another short to medium documents, would deliver better performance if they could be combined properly. Finally, we investigated the problem of selecting documents for relevance feedback. Unhappy with the assumption that all of the top 20 retrieved documents, for example, are relevant and ready for a relevance feedback process, we revisited the cluster hypothesis [2] and experimented with clustering the top 20 documents and automatically selecting a subset for relevance feedback.

Our research system named EUREKA (End User Research Enquiry and Knowledge Acquisition) was used for carrying out the experiments. EUREKA consists of a rich set of UNIX tools which can be assembled into various automatic indexing and ranking/filtering mechanisms, either as a new retrieval system or as a simulation of an interesting research system. The tool set design provides a maximum level of flexibility.

The remaining document is organized as follows: Section 2 describes a strategy for processing short NL queries and reports experiment results. Section 3 describes a strategy for data fusion and presents related experimental results. Section 4 describes a selective relevance feedback process and discusses related experimental results. Note that every experiment reported in these sections used the TREC4 ad hoc data and queries--our training materials for preparing for TREC5. Section 5 summarizes the training work. And finally, Section 6 comments on our TREC5 results.

2.0 Strategy for Processing Short NL Queries

The characteristics of short NL queries include 1) one sentence or one incomplete sentence, 2) lack of informative terms, and 3) lack of term frequency. The lack of query terms leads to an increased level of query ambiguity, and the ambiguity in turn tends to lead to a deterioration of retrieval performance. On the other hand, the lack of term frequency information is detrimental to a statistical oriented retrieval system. The combination of these two factors could make the performance of an IR system deteriorate in the neighborhood of 30 percent [3]. Any strategy for processing short NL queries, to be effective, needs to reduce the level of ambiguity as well as to compensate for the loss of query term frequency.

One strategy proposed in [4] focuses on reducing the level of query ambiguity through a set of Boolean constraints. The strategy demands that searchers are capable of grouping the informative terms in a short NL query into subtopic sets (i.e., concepts or themes). The conjunction constraints in general require the retrieved documents to have most or all of the subtopic sets. The constraints can be implemented as a filtering mechanism on the top of a ranking system.

The Boolean constraints made a noticeable improvement to the short version of theTREC2 and TREC3 queries, but produced a mixed result on the short TREC4 queries. The reported improvement, however, may not be entirely attributed to the Boolean constraints due to the fact a number of the queries received additional terms during the manual construction of the queries. Any addition of quality terms to a short NL query could have a positive impact on its retrieval effectiveness. Also the requirement of manual construction of Boolean constraints makes this strategy less likely to be adopted as a real implementation.

Another strategy reported in [5] focuses on compensating for the loss of query term frequency in short NL queries. The strategy attempts to detect "significance" of each query term using several collection statistics such as average term frequency within a document and inverse term document frequency with a cut-off. The strategy made a significant improvement in retrieval effectiveness both in the original query runs as well as in the expanded query runs. Because the process is automatic, its chance for real application is great. It may be worthwhile to point out that the true average term frequency within a document is an expensive measurement, especially given a huge document collection; various estimations of the average, on the other hand, can be made with different confidence levels.

We have developed a different strategy for tackling both the problem of query ambiguity and the problem of lack of query term frequency. The assumption is that nouns and noun phrases, other than those popular ones, usually represent important concepts in short NL queries, and their inter-term distances in the queries approximate their conceptual distances [4, 6,7]. Taking TREC query 203, for example,

> What is the economic impact of recycling tires?

the noun "economic" and the noun "recycling" are two words apart and their conceptual distance is "longer" than that between "recycling" and "tires". But it is debatable whether the conceptual distance between "economic" and "tires" is farther than that between "economic" and "recycling."

Taking TREC query 207 as an another example,

> What are the prospects of the Quebec separatists achieving independence from the rest of Canada?

the noun "Quebec" and the noun "separatists" are closer to each other than "Quebec" and "Canada", while "Quebec" and "Canada" are "equally" close to the noun "independence. As in many natural language analyses, one can find many supportive cases and one can also easily discover some negative situations. With an application in our mind, we are happy with a plausible assumption. From this assumption we hypothesize that:

- The further apart a pair of nouns appear in short NL queries, the further distinct the two represented concepts are from each other, and therefor the more desirable that the two concepts co-occur in retrieved documents (That is, a softer constraint than the Boolean conjunction [8]).
- And this 2-noun dependence expressed in a function of inter-term distance can improve the effectiveness of retrieving documents for short NL queries.

These hypotheses should not be confused with the theory of probabilistic term dependence usually expressed in the form of Bahadur Lazarsfeld expansion [9]. The hypotheses are only associated with the *initial* retrieval run before a relevance feedback run, and are not yet related to any estimation of probability parameters for a given set of relevant documents and a given set of non-relevant documents. In addition, the hypotheses only consider 2-noun dependence. The goal is to retrieve much more relevant documents at the top of a rank list so that a relevance feedback process, whether it is a probability based or the vector based, can be more effective. Another practical goal is to develop a handle with which one can have a certain degree of control on the quality of top-ranked documents.

To test the hypotheses, we selected the rank list from Cornell's Lnu.ltu run as our benchmark [10], the best performer in handling the short TREC4 queries, and we also developed two ranking functions, one derived from Okapi BM25 [11] for favoring medium to long documents (M-L function for short) and another derived from Lnu.ltu [10] for favoring medium to short documents (M-S function). Some characteristics of these two functions are described in Table 1 and 2. Both functions can give a document a bonus score when they have a match to a 2-noun dependence specification. The bonus score is calculated using both the weights assigned to the two nouns and an inverse measure of their inter-term distance.

Function	**Ave Ret Doc Length (byte)**	**Length Range**
M-L	4,125	404 - 733,919
M-S	4,069	356 - 149,009
Lnu.ltu	4,664	388 - 252,888

TABLE 1. Length Characteristics of the Retrieved Documents

Function	Ret_Doc	Ret_Doc_Overlap	Ret_Rel_Doc	Ret_Rel_Doc_Overlap
M-L	49000	37,115	3634	3,407
M-S	49000		3665	
Lnu.ltu	49000	28,298	3709	3,126
M-S	49000		3665	

TABLE 2. Overlap of the Retrieved documents from the 3 functions

There are two types of test measures. The precision biased measures are the precision ratios at the six cut-off points among top 20 documents, namely, 1st, 3rd, 5th, 10th, 15th and 20th. These precision ratios allow us to closely observe the behavior changes of the two ranking functions. On the other hand, the recall biased measure is merely the number of relevant documents retrieved among the 49,000 retrieved documents for a TREC4 ad hoc run. This number has a strong correlation to the 11-point average precision measure but is more readable. Table 3 describes the test results. The positive percentages in parentheses are the ratios of improvement and the negative percentages are the ratios of deterioration, in comparison to the baseline. The punctuation ":" in the table headings separates the experimental conditions.

System	Lnu.ltu (baseline)	M-L	M-L;2-noun-dependence	M-S	M-S; 2-noun-dependence
Precisions					
1st	.53	.57 (8%)	.69 (30%)	.57 (8%)	.57 (8%)
3rd	.55	.58 (5%)	.63 (15%)	.52 (-6%)	.60 (9%)
5th	.51	.54 (6%)	.58 (14%)	.55 (8%)	.59 (16%)
10th	.46	.47 (2%)	.51 (11%)	.50 (9%)	.53 (15%)
15th	.44	.45 (2%)	.46 (5%)	.46 (5%)	.48 (9%)
20th	.41	.43 (5%)	.44 (7%)	.42 (2%)	.45 (10%)
Recall					
rel-doc-ret	3709	3634 (-2%)	3681 (-1%)	3665 (-1%)	3694 (0%)

TABLE 3. Query term dependence experiments

The results in Table 3 suggest that both M-L function and M-S function can perform at the same level of effectiveness as Lnu.ltu function does, if not at a better level in terms of precision. The introduction of 2-noun dependence to M-L function and M-S function does improve the effectiveness further. The improvements may not be statistically significant when compared to their base performance (M-L and M-S), but the precision improvements over the baseline Lnu.ltu may be significant. The recall measure has been held steady in these experiments. The results support our hypotheses. In addition, we think that further improvement is achievable.

3.0 Strategy for Data Fusion

With the two performing ranking functions (M-L and M-S), we turned our attention to a data fusion strategy to further improve the retrieval effectiveness reported in Table 1. The data fusion approach is based on the phenomenon that different ranking functions tend to retrieve a different set of documents, and, more specifically, a different subset of relevant documents. A good fusion mechanism can retrieve a set of relevant documents that is larger than every individual subset of relevant documents from the individual ranking functions [1]. The most important feature is the ability to retrieve *different documents*. A list of ranking functions from the same family such as SMART [12], on the other hand, may share more commonalities than differences--making a fusion process less meaningful. This, in fact, was one of our motivations to develop the specialized ranking functions.

To emphasize the diversity of the ranking functions in our data fusion tests, we decided to add another function which ranks documents based entirely on their best local relevance scores [13, 14]. We defined a local text window as a list of query term occurrence(s) in a span of 250 words (including stopword positions) from a given query term. The number of text windows that a document has is determined by the number of occurrences of a given set of query terms. We were initially thinking of using the traditional cosine function [15] since it strongly favors short documents. But we actually used M-L function (M-L-LOC) because of the fact that the window has a fixed size.

We experimented with a few fusion methods and finally selected Logistic Regression [16]. The input to the regression analysis is a table consisting of eight fields, namely, binary relevance indicator, weighted publication indicator, the relevance score and the rank from M-L function, the relevance score and the rank from M-S function, and the relevance score and the rank from M-L-LOC function. The field weighted publication deserves additional description. This field can accommodate the information such as an estimated relevant document distribution among different publications or data sources (e.g., AP and FR), or user’s preference of publications (e.g, Patent better than ZIFF). However, in preparing for the TREC5 runs we turned this field off by assigning a constant to it, since we did not have any knowledge about the data on Disk4 at that time.

The testing plan was to use the data on Disk2 as the training data and the data on Disk3 for testing. But due to time constraints, we used the both disks in training, and later applied the trained fusion function to the TREC5 ad hoc runs. As a result, we describe the training results in Table 4 and defer the discussion of the testing in the coming TREC5 section.

In Table 4, columns Lnu.ltu, M-L and M-S are copied from Table 3, and the baseline for comparisons is still Lnu.ltu. Column M-L-LOC covers the results from the local text window run. It is not surprising that this run has a serious deterioration from the baseline. The documents retrieved in this run are different from those from M-L and M-S, especially those long documents with isolated relevant texts. The fusion training results are positive both in the precision measures and in the recall measure. The generalizability of this training will be discussed in the TREC5 section.

System	Lnu.ltu (baseline)	M-L	M-S	M-L-LOC	Fusion
Precisions					
1st	.53	.57 (8%)	.57 (8%)	.53 (0%)	.63 (19%)
3rd	.55	.58 (5%)	.52 (-6%)	.51 (-7%)	.63 (15%)
5th	.51	.54 (6%)	.55 (8%)	.48 (-6%)	.61 (20%)
10th	.46	.47 (2%)	.50 (9%)	.43 (-9%)	.52 (13%)
15th	.44	.45 (2%)	.46 (5%)	.40 (-9%)	.48 (9%)
20th	.41	.43 (5%)	.42 (2%)	.38 (-7%)	.46 (12%)
Recall					
rel_ret	3709	3634 (-2%)	3665 (-1%)	3401 (-9%)	3963 (7%)

TABLE 4. Data Fusion Training Results using the TREC4 Data

4.0 Strategy for Selective Relevance Feedback

Most systems involved in TRECs have adopted a 2-step approach in performing the ad hoc task: retrieve an initial set of, say, the top 20 documents for each test query, then feed the initial set into a relevance feedback process to retrieve the final rank list for that query. Usually these systems assume that every document in the initial set is relevant for the sake of automation. The flaw in this assumption is obvious--there are non-relevant documents in the initial set. If we could select relevant documents or remove non-relevant documents automatically, we would expect better performance from relevance feedback.

To this end, we deduced the following guidelines from the cluster hypothesis [3]:

- Both the relevant and the non-relevant documents in an initial set have a tendency towards grouping because they are all highly selected for a common query.
- The relevant documents, however, have a stronger tendency towards grouping than the non-relevant documents.
- In addition, the observation that every document in an initial set is clustered into one group usually indicates that the initial set contains many, if not all, relevant documents, and therefore a relevance feedback process will benefit most from this set.
- To generalize the above further, one large, stand-alone cluster after clustering has a large quantity of relevant documents. Detecting and extracting this cluster is easy.
- On the other hand, the observation that every document in an initial set is a cluster by itself (i.e., all singletons) probably suggests that there is no relevant document in the set, and therefore no relevance feedback should be initiated after clustering.

- Most singletons or orphan documents after clustering are non-relevant documents, and can be removed automatically.

Topic 207, perfect
Grp 1: 106458.r 106783.r 107790 110287.r 114271 117292.r 121838.r 123578.r 133508.r 146688.r 173399.r 269603.r 271890.r 278827.r 278926 279092.r 280452 306156.r 398792.r 289861.r
Topic 209, good
Grp 1: 000403 000434 000513.r 057655.r 066482.r 242256 243847 245128.r 245548 246061.r 246449 248076.r 248747.r 251392.r 286365
Grp 2: 044832 076895
Grp 3: 069711
Grp 4: 090430
Grp 5: 390461
Topic 232, bad
Grp 1: 015128
Grp 2: 025152
Grp 3: 048788
Grp 4: 079357
Grp 5: 143456
Grp 6: 153009
Grp 7: 241340
Grp 8: 260543
Grp 9: 269575
Grp 10: 290721.r
Grp 11: 292551
Grp 12: 304266
Grp 13: 312705
Grp 14: 315733.r
Grp 15: 345581 345879
Grp 16: 363641
Grp 17: 387106
Grp 18: 477459 486460

TABLE 5. Examples of Top 20 Document Clustering (TREC4)

Some examples based on the above guidelines are shown in Table 5, ranging from perfect to bad. The 6-digit document ids are from EUREKA and the small suffix letter “r” indicates a relevant document. The clustering process that generated the examples was a one-pass, flat grouping process. The similarity matrix used came from a variant of the cosine similarity function. A similarity threshold was selected for grouping the top ranked documents.

We assessed the reasonableness of the deduced guidelines.Table 6 contains a few sets of comparisons from the assessment. There are four similarity thresholds corresponding to the four singleton columns on the right. The thresholds are in an increasing sequence to make increasingly more but smaller clusters. The second threshold is also responsible for column “Largest Clust2.” The baselines are the average precision of the 49 sets and the number of the sets that do not have any relevant document.

These two measures together provide a balanced view on the change of the 49 sets, since the increased precision leads to an increased number of sets that do not contain any relevant document. To minimize the number of sets consisting of all non-relevant documents as well as to maximize the density of relevant documents in the sets, we selected the strategy of removing the singletons from a clustering with a lower similarity threshold, instead of the strategy of selecting the largest clusters.

	Initial Sets	**Largest Clust2**	**Singleton1**	**Singleton2**	**Singleton3**	**Singleton4**
Ret_Docs	980	478	756	594	449	343
Ret_Rel_Docs	441	283	380	321	256	198
Ave_Precision	.45	.59	.52	.54	.57	.58
No_RelDoc_Sets	1	10	4	4	8	12

TABLE 6. 2 Different Approaches to Improving the Initial Set for Relevance Feedback

In Table 7 are the test results using the improved sets from “Singleton2”. We used one of the relevance feedback functions in EUREKA. The function was assisted with and without the clustering processing. The symbol “RF” stands for relevance feedback, the symbol “clust” for clustering. The punctuation “;” separates the different ranking processes. The results show a positive contribution from the clustering analysis, but the contribution may not be significant in this test. The results are not conclusive.

System	**Lnu.ltu**	**Lnu.ltu; RF**	**M-L;2-noun-dependence**	**M-L;2-noun-dependence;RF**	**M-L;2-noun-depndnce;clust;RF**
Precisions					
1st	.53	.55 (4%)	.69 (30%)	.65 (23%)	.61 (15%)
3rd	.55	.53 (-4%)	.63 (15%)	.56 (2%)	.56 (2%)
5th	.51	.52 (2%)	.58 (14%)	.56 (10%)	.58 (14%)
10th	.46	.51 (11%)	.51 (11%)	.53 (15%)	.56 (22%)
15th	.44	.49 (11%)	.46 (5%)	.51 (16%)	.52 (18%)
20th	.41	.46 (12%)	.44 (7%)	.48 (17%)	.49 (20%)
Recall					
Rel_Ret	3709	4350 (17%)	3681 (-1%)	4219 (14%)	4335 (17%)

TABLE 7. Selective Relevance Feedback Experiment

5.0 Summary of our TREC4 Training

The strategy that applies 2-noun dependence in ranking seems help in improving the quality of top ranked documents. The improvement, however, is not dramatic. There is further potential for this indexing device. For example, by combining the query term selection strategies proposed in [6] we could make this device more precise. Furthermore, this device could be introduced into the probabilistic models in [17, 18] as part of the query networks.

The strategy that employs different or specialized ranking functions in data fusion seems help in improving overall retrieval effectiveness. But our fusion function trained with the TREC4 data is subject to the TREC5 test.

Being selective as to the top ranked documents in a process of relevance feedback seems help in improving retrieval effectiveness in general. The test results reported, however, stopped short of the 10% improvement margin, the margin usually implies a statistically significance. Perhaps with a better tuned clustering algorithm, the 10% improvement is attainable.

6.0 TREC5 Results

We submitted two runs, LNaDesc1 and LNaDesc2, for the required short query test (i.e., using the description section only). LNaDesc1 came from the trained TREC4 fusion function. The fusion combined the 3 rank lists from M-L function, M-S function, and M-L-LOC function. M-L function and M-S function were assisted with the 2-noun dependence device as well as the selective relevance feedback. The rank list from M-L function, also assisted with the 2-noun dependence device and the selective relevance feedback, was submitted individually as LNaDesc2, due to the uncertainty surrounding the new data on Disk4 and over the generalizability of the fusion function. Submitting the M-L rank list rather than the M-S list was due to the concern that its parent function Lnu.ltu might have been over fitted to the TREC4 materials.

The fact that LNaDesc2 performed better than LNaDesc1 in TREC5 indicates that the TREC4 fusion function suffered from over-fitting to the TREC4 data. But the performance deterioration is marginal. We will look into our TREC5 results after the conference.

The other two ad hoc submissions, LNmFull1 and LNmFull2, were incomplete runs. The original plan for the two manual runs was to use a statistical thesaurus [3] compiled from the data on Disk 3 and 4 to construct the test queries. Unfortunately the statistical thesaurus did not get compiled on time for the test runs, forcing us to abandon that plan.

Instead, we re-ran the processes for generating LNaDesc1 and LNaDesc2 with the addition of the manually edited full queries. For these two runs, we did not use the 2-noun dependence device because it might get "disoriented" in long queries. Finally, we did not perform the clustering of the top ranked documents for these two runs. Our motivation for submitting these runs is to establish baselines for our future research. For example, we have been thinking of the process in which a human subject is given an opportunity to manually select one or more document clusters for manual relevance feedback [19]. We may just do that for the final conference paper.

Acknowledgments: The authors of this note would like to thank Daniel Pliske, David Miller and Vladimir Nayfeld for their reviewing work.

REFERENCES:

[1] Belkin, N. J., Kantor, P., Cool, C. and Quatrain, R. "Combining evidence for information retrieval," The Second Text REtrieval Conference (TREC2), NIST Special Publication 500-215, pp.35-44, edited by D. Harman, 1994.

[2] van Rijsbergen, C. J. Information Retrieval. Butterworths, London, 1979.

[3] Lu, X. A. and Keefer, R. B. "Query expansion/reduction and its impact on retrieval effectiveness," The Third Text REtrieval Conference (TREC3), NIST Special Publication 500-225, pp.231-239, edited by D. Harman, 1995.

[4] Hearst, M. A. "Improving full-text precision on short queries using simple constraints," Proceedings of 5th Annual Symposium on Document Analysis and Information Retrieval, pp.217-232, Las Vegas, Nevada, 1996.

[5] Kwok, K. L. "A new method of weighting query terms for ad-hoc retrieval," In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.187-196, Zurich, Switzerland, 1996.

[6] Keen, E. M. "The use of term position devices in ranked output experiments," Journal of Documentation, Vol.47, pp.1-22, 1991.

[7] Searle, J. R. Speech Acts. Cambridge University Press, London, 1984.

[8] Salton, G., Fox, E. A. and Wu, H. "Extended Boolean information retrieval," Communication of the ACM, Vol.26, pp.1022-1036, 1983.

[9] Yu, C. T., Buckley, C., Lam, K. and Salton, G. "A generalized term dependence model in information retrieval," Information Technology: Research and Development, Vol.2, pp.129-154, 1983.

[10] Singhal, A., Salton, G. Mitra, M. and Buckley, C. "Document length normalization,", Information Processing & Management, Vol.32, pp.619-633, 1996.

[11] Robertson, S. E., Walker, S. Jones, S., Hancock-Beaulieu, M. M. and Gatford, M. "Okapi at TREC3," The Third Text REtrieval Conference (TREC3), NIST SPecial Publication 500-225, pp.109-126, edited by D. Harman, 1995.

[12] Fox, E. A., Koushik, M. P., Shaw, J. Modlin, R. and Rao, D. “Combining evidence from multiple searches,” The First Text REtrieval Conference (TREC1), NIST Special Publication 500-207, pp.319-328, edited by D. Harman, 1993.

[13] Buckley, C., Salton, G. and Allan, J. “Automatic retrieval with locality information using SMART,” The First Text REtrieval Conference (TREC1), NIST Special Publication 500-207, pp.59-72, edited by D. Harman, 1993.

[14] Callan, J. P. “Passage-level evidence in document retrieval,” In Proceedings of the 17th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.302-310, Dublin, Ireland, 1994.

[15] Salton, G. and McGill, M. J. Introduction to Modern Information Retrieval, McGraw-Hill Book Co., NY, 1983.

[16] Cooper, W. S., Chen, A. and Gey, F. C. “Full text retrieval based on probabilistic equations with coefficients fitted by Logistic Regression,” The Second Text REtrieval Conference (TREC2), NIST Special Publication 500-215, pp.57-66, edited by D. Harman, 1994.

[17] Turtle, H. R. and Croft, W. B. “Evaluation of an inference network-based retrieval model,” ACM Transactions on Information Systems, Vol.9, pp.187-222, 1991.

[18] Kwok, K. L. “A network approach to probabilistic information retrieval,” ACM Transactions on Office Information Systems, Vol.13, pp.325-353, 1995.

[19] Hearst, M. A. and Pedersen, J. O. “Reexamining the cluster hypothesis: Scatter/Gather on retrieval results,” In Proceedings of the 19th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp.76-84, Zurich, Switzerland, 1996.