Using Multiple External Collections for Query Expansion

Dongqing Zhu and Ben Carterette
Department of Computer & Information Sciences
University of Delaware
Newark, DE, USA 19716
[zhu | carteret]@cis.udel.edu

1. INTRODUCTION

For the 2011 Medical Records Track, we used several external collections for query expansion and mainly explored three research questions:

First, we investigated the possibility of using query sessions from PubMed query logs for improving the estimation of a relevance model. In a typical search scenario, a user may submit multiple queries before she actually finds satisfactory search results. These closely related queries form a single query session which represents a single information need. By finding relevant query sessions with regard to a Medical Track topic we can incorporate into our relevance model useful query terms which reflect real information needs that are more or less related to the Medical Track topic.

Second, we explored how the size and quality of external collections would impact the effectiveness of query expansion. More specifically, we used TREC 2007 Genomics Track data and ImageCLEF 2009 Medical Retrieval data. The former collection is more genomics-related and is larger while the latter one is more medical-related and is much smaller. Intuitively, it is more likely for a larger external collection to contain more good expansion terms. However, the quality (in terms of the overlapping concepts between the target collection and an external collection) can be an important factor as well. This allowed us to carry out a pilot study on the relationship between collection quality, size, and the effect on query expansion.

Third, we used a mixture of external collections for query expansion. In particular, we explored methods that can adaptively combine evidence from multiple collections for different topics. Usually, the weights for a mixture relevance model are determined via training on a test collection, and thus are fixed across all topics. If we could estimate the concept overlapping of a topic with external collections and assign weights for the mixture model accordingly, the system can be adaptive to topics and may achieve a better performance. That is the motivation for this third research direction.

We first describe our retrieval models and systems in Sections 2 and 3. Then in Section 4 we show and compare the official TREC evaluation results of our submissions, and further analyze our retrieval system performance based on the test collection. Following that, we discuss the above research questions in Section 5. We conclude in Section 6.

2. RETRIEVAL MODEL

We used a language modeling-based approach of applying a mixture of relevance models for query expansion as described by Diaz and Metzler [3]. The expanded relevance model estimate based on the original query and an external collection was implemented in the Indri[1] system by formulating a query in the following format:

#weight(λ #combine(w_1 w_2 ... $w_{|Q|}$) $(1-\lambda)$ #weight(p_1 e_1 p_2 e_2 ... p_m e_m)),

where λ is the weight assigned to the original query language model, w's are terms from the original query, and e's are the m expanded terms with the highest probabilities p's which are computed by the formula:

$$p_i = p(e_i|\hat{\theta}_Q) = \sum_{j=1}^{k} p(e_i|\theta_{d_j}) p(Q|\theta_{d_j}), \quad (1)$$

where $\hat{\theta}_Q$ is the estimate of relevance model based on an external collection, d_j's are top-ranked k documents retrieved from the external collection, and θ_{d_j} is the document language model of d_j. An expanded query looks like the following:

#weight(0.7 #combine(female breast cancer mastectomies admission) 0.3 #weight(0.225 mastectomy 0.145 women 0.110 risk 0.107 prophylactic 0.101 bct 0.074 radiate 0.068 therapy 0.062 radiotherapy 0.058 surgery 0.050 adjuvant))

This Indri query format can be extended to use multiple external collections for query expansion by formulating a #weight expression for each collection separately, then including them in the Indri query with a new weight parameter (such that weight parameters always sum to 1).

[1] http://lemurproject.org/indri/

3. RETRIEVAL SYSTEMS

This section describes six systems. We submitted 4 runs based on these systems. The first system used the target collection (i.e., medical records collection) only. The rest all used external information. We implemented all systems using Indri and trained them on the TREC sample test collection which contains 4 sample queries and 27 relevant visits. In addition to the standard stopwords, we also removed 'patient' and 'patients' in the topics because they are common words in the medical records.

3.1 Baseline

The baseline system used the target collection only. However, before indexing we merged multiple reports from the same visit into one single visit file based on the report-to-visit mapping information provided by NIST, which converted 100,866 reports to 17,198 visit files. In the retrieval process, the Dirichlet smoothing parameter μ of the language model was the only free parameter for this baseline system and was trained on the sample test collection.

3.2 Using Diagnosis Description

For this system, we further expanded all visit files by replacing the admission and discharge diagnosis codes with their corresponding descriptions[2] (note that this procedure was taken before indexing for all the following systems as well). In addition, since the patient de-identifying procedure marked the age entities in a systematic way across all reports, we also extracted the age information, if there was in the report, and made it as a new field. The retrieval process was exactly the same as the previous system.

3.3 Using Genomics Data

This system used the TREC 2007 Genomics Track dataset for relevance model estimation. This dataset contains 162,259 full-text articles in HTML format from 49 genomics-related journals [4]. We did not pre-process this dataset and we used Indri's default setting for retrieving documents from Genomics data. The expanded query model was computed according to Equation 1. Parameters λ, m, and k were trained by sweeping them over their parameter spaces.

3.4 Using ImageCLEF Data

Another external collection is the ImageCLEF 2009 medical image retrieval dataset which contains 74,902 images from two radiology journals, Radiology and Radiographics. Each image has a corresponding XML file containing its caption and article title [7]. The file also contains a URL of the article in which the image appears. This allowed us to crawl an additional 5,704 full-text articles as another external collection. We followed the same procedure as the previous system for estimating the query model. In this paper, we denote the collection containing captions and titles as CLEF-CT and the one containing full-text articles as CLEF-A.

3.5 Using PubMed Query Log

In this system, a one-day PubMed query log was used. The users in the log are de-identified by the NLM to protect their privacy [5]. The content of the log file looks like the following:

```
YAAAAI|63|alzheimer's disease inhibition
kva2Y4IOFlsAAAx3xiYAAAAM|63|RNA interference plymerase
...
4AAAAH|103|breast cancer and insulin resistance
jFnDXYIOFpIAAEAb0QAAAAAH|103|Trpm5 insulin
...
08AAAAF|252|intracerebral hemorrhage
2ZK4IOFkQAABdJCyAAAAAJ|252|carotid artery track
3JNvrYIOFl0AAHs3MNoAAAAJ|252|papain AND teeth
```

Each line in the log file contains three parts, namely an anonymized user ID, seconds since midnight EST, and an query issued by that user, which are separated by vertical bars. There are 2,996,301 queries submitted by 627,455 unique users within a day, from midnight to midnight, and there is no click-through information associated with these queries.

After excluding users who are considered as 'bots' (who submitted over 50 queries per day) and all the null queries, we had 2,657,316 queries from 611,083 users. Figure 1 shows the percentage distribution of number of users who issued a specific number of queries within a day. About 66% of the users submitted more than 5 queries. We further removed all special symbols, punctuations, and logical operators such as AND and OR from each query.

According to our hypothesis aforementioned, it is desired that query sessions could be identified from each *user session* (which contains all the queries submitted by a single user). However, finding query session boundaries is itself an open research question. Usually a time window of 30 minutes is used to separate sessions. There are other methods proposed specifically for query session segmentation in this one-day PubMed query log, such as using semantic and contextual information [5, 6]. For simplicity, we used a similar approach based on time for identifying query session boundaries as described below:

1. Treat each PubMed *query* (made italic in all the following steps to disambiguate it from the original medical records queries) as a document and index the query log.
2. For each of the k top-ranked *queries*, assign 1.0 as the weight for all query terms.
3. Obtain the corresponding user session of each top-ranked *query* (denoted as Q). All other *queries* in that user session are considered as relevant to the topic as well. However, the query term weights for those queries decay exponentially as a function of how far (in seconds) they are away from Q in that user session. If there are multiple *queries* found relevant within a single user session, the weights of other *queries* in that session will be computed according to their closest Q's.
4. Aggregate term weights to get the top weighted m terms and compute $p(e_k|\theta_d)$'s in Equation 1.

The assumption behind the above approach is that it is more likely for a user to submit two related queries if those two

[2]We crawled diagnosis code descriptions from `https://drchrono.com/public_billing_code_search`

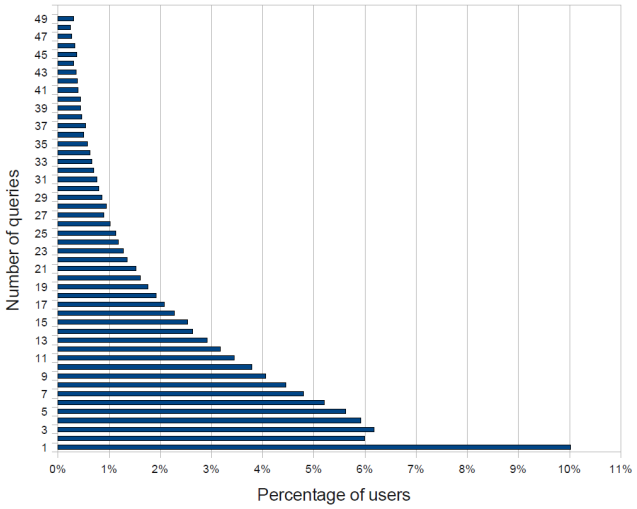

Figure 1: Distribution of number of users over number of queries they submitted.

queries are close in time. Thus, we were essentially determining relevant *query sessions* by using a soft boundary, which was implemented as a decaying function.

3.6 Using Two External Collections

Results on the sample test collection indicated that the Genomics data and PubMed query Log improved the baseline result by 20% to 30%. However, the ImageCLEF data did not show improvement. Thus, we used a mixture of the two promising datasets for our last system.

4. RESULTS AND ANALYSIS

We submitted four runs to TREC based on the experimental results on the sample test collection. We summarize and analyze the results in this section.

4.1 TREC Results

Table 1 gives a summary of the runs from the six systems described in Section 3. We selected four runs for TREC submission as indicated in the last column. Table 2 shows the official evaluation results of the 4 submissions on four major evaluation measures. Results are based on the top 1000 retrieved visits for each run. *udelgn* has the best overall performance among the four runs.

All our four runs were in the second round of TREC submissions and thus they were not pooled and judged. Thus, we only compare our runs with the unjudged group. *udelgn* and *udelbl* are above the median for the majority of topics on all three official evaluation measures (P@10, bpref, R-prec), while *udelgn* and *udelbl* are below the median for about half of the topics. Figure 2 shows the result comparison for R-prec.

4.2 Analysis

Based on the evaluation results and our initial motivation of using external collections, we want to address several questions as listed below:

No.	External Resource	RunID	Submitted
1	none	udelba	
2	DCD (Diagnosis Code Desc.)	udelbl	√
3	DCD + Genomics	udelgn	√
4.a	DCD + CLEF-CT	udelcf	
4.b	DCD + CLEF-A	udelcfa	
5	DCD + Query Log	udelpm	√
6	DCD + Genomics + Log	udelmx	√

Table 1: Run Summary

SystemID	P@10	bpref	R-prec	MAP
udelbl	0.5324	0.5073	0.3907	0.3780
udelgn	0.5441	0.5217	0.4068	0.3924
udelpm	0.4206	0.4201	0.3085	0.2729
udelmx	0.4382	0.4545	0.3240	0.2827

Table 2: Official Evaluation Results (averaged over 34 topics)

1) How much improvement over the baseline did we actually get after replacing the diagnosis codes with the description?

2) Are ImageCLEF data useful though we did not use them for TREC submission?

3) Why are the results of *udelpm* and *udelmx* worse than the other two runs? Does that mean the one-day PubMed query log is not applicable for our retrieval model?

To answer these questions, we re-evaluated all the systems described in Section 3 using cross-validation. The reason for using cross-validation is that we previously trained our systems on a very small sample test collection which might not reveal the true system performance well. Thus, we used 5-fold cross-validation on the official test collection. In each iteration, the retrieval model was trained on 28 queries to obtain the parameter setting for the best mean average precision (MAP) by sweeping over the parameter spaces according to Table 3. Then the trained model was used to generate a rank list for the remaining 7 queries. After this process, we could get five separate rank lists which were further merged into one containing results for all the 35 topics (Topics 130 was dropped by TREC and actually not used for training here). Finally, we evaluated the merged results.

Table 4 shows results of cross-validation. By comparing *udelba* and *udelbl*, we can see that replacing diagnosis codes with their descriptions improves the baseline by 4% to 6% across all 4 evaluation measures. Using the Genomics data

Parameter	From	To (Exclusive)	Step Size
μ	1000	30000	1000
λ	0.0	1.0	0.1
k	5	60	5
m	10	50	10

Table 3: Parameter space for training. μ is the Dirichlet smoothing parameter, λ is the collection weight, k is the number of top-ranked documents, and m is the number of expansion terms.

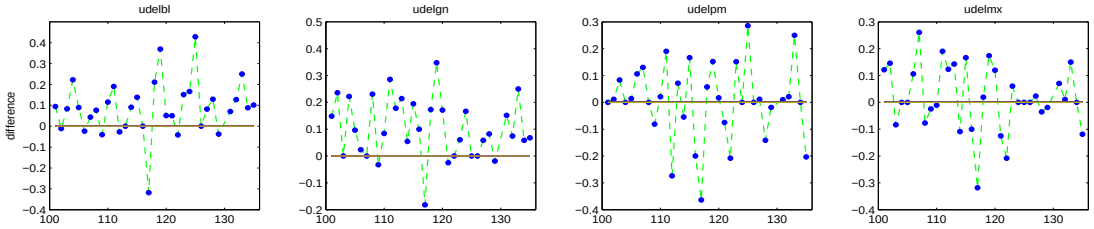

Figure 2: Difference between each system and median of TREC R-prec of the 80 unjudged runs for all 34 topics. Systems *udelgn* and *udelbl* are above the median for the majority of topics, while systems *udelgn* and *udelbl* are below the median for about half of the topics. Results of P@10 and bpref are similar to R-prec.

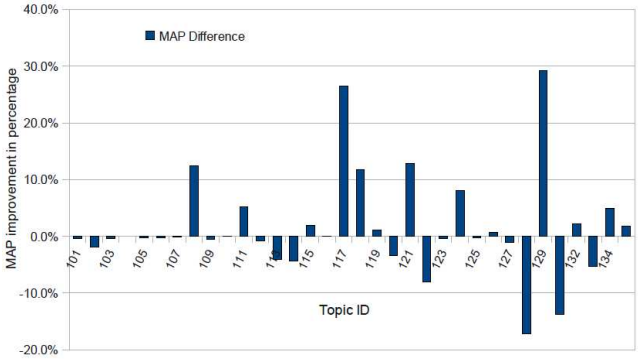

Figure 3: MAP improvement of *udelpm* relative to *udelbl*. System *udelpm* performs relatively better than system *udelbl* on $7 \sim 8$ topics and relatively worse on about the same number of topics. For the majority of topics, these two system have roughly the same performance.

for query expansion further improves the baseline by 4% on MAP. System *udelmx* has the best MAP though the training time is relatively longer than the other systems.

If we compare *udelcf* and *udelcfa* with *udelbl*, *udelcf* is just slightly better than *udelbl* on *bpref*. Thus, using CLEF-CT actually hurts the system performance. However, *udelcfa* consistently improves the baseline by more than 10% across all evaluation measures. This indicates that CLEF-A is better than CLEF-CT for query expansion.

Furthermore, cross-validation results of systems *udelpm* and *udelbl* are quite different from their official TREC evaluation results. We think it was simply because the retrieval models of *udelpm* and *udelmx* overfitted the small sample test collections at that time. However, it seems that using the PubMed query log for query expansion has little effect on the system performance. We suspect that it is because the PubMed query log is quite different from the other external collections. For instance, queries are short but vary in length and format across different users. Typos are common in the log. Also, the log contains many navigational queries (e.g. author names, years, PMIDs) that are not useful. In fact, all these factors may prevent us from selecting the good expansion terms in the log. Moreover, the query log is just a one-day log, which means it may not cover all Medical Track topics well. That might be the reason that we observed improvements over a few topics but for the other topics PubMed log degraded the system performance (Figure 3 explains this situation). As a result, the method of using query session for query expansion appeared to be less effective than the other methods. Thus, we need to find a better way to analyze this query log and extract useful information from it.

5. FURTHER EXPLORATION

In this section we describe some pilot studies. Based on the analysis in Section 4.2 we want to further explore two problems: 1) How the quality and size of the external collection may impact the performance of our retrieval systems; 2) How to effectively combine multiple external collections for query expansion. Some pilot studies are describe below.

5.1 Quality vs. Size

The Genomics corpus is more genomics-related and is larger while the ImageCLEF corpus is more medical-related and is much smaller. It is more likely for a larger external collection which has overlapping concepts with the target collection to retrieve good expansion terms than a smaller one [2, 8, 3]. However, the quality of collections, in terms of their similarity with the target collection, is also an important factor [3]. For comparison, we summarizes the statistics of all external collections in Table 5.

The CLEF-A collection is a superset of the CLEF-CT and thus is better in quality and size. Though Genomics collection may not be as good as both ImageCLEF collections in terms of quality, it is magnitudes larger than both ImageCLEF collections. That might be the reason that system *udelgn* outperformed *udelcf* and is only slightly inferior to *udelcfa* as shown in Table 4.

5.2 Using Multiple Collections

Different external collections may all improve an initial ranking for a specific query but in different ways (e.g., Topics 129, 132, and 135 in Figure 4). Also, some external collections may improve an initial ranking while others may hurt the same initial ranking (e.g., Topics 124, 125, and 128 in Figure 4). That explains why system *udelmx* has the best MAP performance. However, the training time of this kind of systems grows exponentially as the number of external collection increases. If we can combine different external collections in a way that they are adaptive to specific queries, we can not only obtain better results than using a single external collection, but also improve the extensibility of the

System ID	External Resource	MAP	P@10	bpref	R-prec
udelba	none	0.3527	0.5059	0.4694	0.3654
udelbl	DCD	0.3741 (+ 6.1%)	0.5265 (+ 4.1%)	0.5004 (+ 6.6%)	0.3901 (+ 6.8%)
udelgn	DCD+Genomics	0.3907 (+10.8%)	0.5500 (+ 8.7%)	0.5057 (+ 7.7%)	0.3931 (+ 7.6%)
udelcf	DCD+CLEF-CT	0.3684 (+ 4.5%)	0.5176 (+ 2.3%)	0.5066 (+ 7.9%)	0.3891 (+ 6.5%)
udelcfa	DCD+CLEF-A	0.3920 (+11.1%)	0.5647 (+11.6%)	0.5188 (+10.5%)	0.4180 (+14.5%)
udelpm	DCD+Query Log	0.3740 (+ 6.0%)	0.5176 (+ 2.3%)	0.5054 (+ 7.7%)	0.3904 (+ 6.8%)
udelmx	DCD+Genomics+Log	0.4007 (+13.6%)	0.5382 (+6.4%)	0.5289 (+12.8%)	0.4146 (+13.5%)
udelcori	DCD+Genomics+CLEF-A+Log	0.3863 (+ 9.5%)	0.5588 (+10.5%)	0.5198 (+10.7%)	0.3981 (+ 8.9%)

Table 4: Results by cross-validation. *udelba* is the baseline system. By using the description of diagnosis codes, system *udelbl* improves the baseline by $4 \sim 6\%$ on all 4 evaluation measures. *udelcf* and *udelpm* perform roughly the same as *udelbl*, while the other systems further improves the baseline. System *udelmx* has the best MAP.

Collection	Documents	Unique Terms	Total Terms	Average Document Length
Genomics	162,259	2,143,156	1,070,113,111	6595
CLEF-A	5,704	162,032	37,050,614	6495
CLEF-CT	74,902	107,482	10,962,310	146

Table 5: Collection Statistics

system. Thus, we borrow the idea of the CORI [1] resource selection algorithm to assign a different set of weights to multiple collections for different queries.

The CORI algorithm is mainly used in distributed information retrieval for resource selection and results merging. The algorithm first obtains information and statistics about individual collections via a query-based sampling approach and builds a set of resource descriptions that can accurately represent the contents of those collections. Then given a query, each collection will get a score based on the resource descriptions. Thus, CORI can select the top-ranked resources to search. Results from multiple resource will then be merged in a new ranking according to both the resource scores and original document scores.

We use CORI to build the resource descriptions for the medical records, Genomics, CLEF-A, and PubMed query log. Then, CORI assigns a different set of collection scores for each topic. We take those scores as collection weights in our relevance model. These scores could be thought of as the 'similarity' measure between a topic and a collection. A collection with high similarity scores may have more overlapping concepts with the topic and thus better expansion terms. We train k (number of top-ranked documents) and m (number of most frequent terms) using 5-fold cross-validation. For simplicity, we use the same set of k and m for all external collections during each iteration of training. The CORI algorithm is already implemented in Indri and we use all the default settings.

We denote this CORI-based system as *udelcori*. The last row of Table 4 shows cross-validation results of *udelcori*. Though *udelcori* is not the top performing system, the results indicate that this weight assigning method is quite promising since it dynamically and automatically determines weights for each expansion clause for each query. Because it is unclear for now how the parameters of CORI will impact the results, we still need to do more exploration in the future.

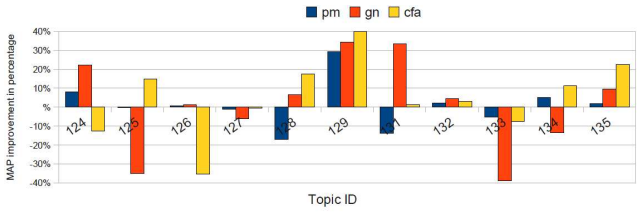

Figure 4: MAP improvement of systems *udelpm*, *udelgn*, and *udelcfa*, relative to *udelbl*.

6. CONCLUSION

We have shown that using medical-related external collections for query expansion can effectively improve the baseline system. In addition, the size and quality of the expansion collection are two key factors of expansion effectiveness, and one can compensate the other. Moreover, the CORI resource selection algorithms can adaptively assign a set of weights to multiple expansion collections as well as the target collection. This query-adaptive resource weighting scheme has shown promising results and is worth further exploration.

7. ACKNOWLEDGMENTS

The authors would like to thank Dr. Zhiyong Lu from NIH, Dr. Jorge R. Herskovic and Dr. Elmer V. Bernstam from University of Texas School of Health Information Sciences at Houston, for providing the one-day PubMed query log.

8. REFERENCES

[1] J. Callan. Distributed information retrieval. In *Advances in Information Retrieval*, pages 127–150. Kluwer Academic Publishers, 2000.

[2] J. Callan, W. B. Croft, L. Ballesteros, D. Byrd, R. Swan, and J. Xu. INQUERY does battle with TREC-6. In *The Sixth Text REtrieval Conference (TREC-6)*, pages 169–206, 1998.

[3] F. Diaz and D. Metzler. Improving the estimation of relevance models using large external corpora. In

Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, SIGIR '06, pages 154–161, New York, NY, USA, 2006. ACM.

[4] W. R. Hersh, A. M. Cohen, L. Ruslen, and P. M. Roberts. TREC 2007 genomics track overview. In *TREC*, 2007.

[5] J. R. Herskovic, L. Y. Tanaka, W. R. Hersh, and E. V. Bernstam. A day in the life of PubMed: Analysis of a typical day's query log. *JAMIA*, 14(2):212–220, 2007.

[6] Z. Lu and W. J. Wilbur. Improving accuracy for identifying related PubMed queries by an integrated approach. *Journal of Biomedical Informatics*, 42(5):831–838, 2009.

[7] H. Müller, J. Kalpathy-Cramer, I. Eggel, S. Bedrick, S. Radhouani, B. Bakke, C. E. Kahn, and W. R. Hersh. Overview of the CLEF 2009 medical image retrieval track. In *CLEF (2)*, pages 72–84, 2009.

[8] S. Walker, S. E. Robertson, M. Boughanem, G. J. F. Jones, and K. S. Jones. Okapi at TREC-6: Automatic ad hoc, VLC, routing, filtering and QSDR. In *TREC*, pages 125–136, 1997.