TREC2005 Robust Track Experiments using PIRCS

K.L. Kwok, L. Grunfeld, N. Dinstl and P. Deng

Computer Science Department, Queens College, CUNY
Flushing, NY 11367

1 Introduction

The two tasks in the TREC2005 Robust track were as follows: given a set of topics, A) predict their ranking according to average precision; and B) improve the effectiveness of their ad hoc retrieval, in particular the weakest topics and if possible with the help of what has been found in task A. The difference from last year is that the test collection (ACQUAINT) is different from the training collection (from TREC2004). The evaluation measures for these tasks have also changed from TREC2004. For task A, it is the difference in area (diff-area) between the observed and the predicted MAP plots when the worst performing topic is successively removed from the set. For task B, it is the GMAP, which is roughly the geometric mean of the average precision of all topics (Voorhees 2005).

We do not believe our techniques for predicting topic average precision is sufficiently accurate to be applied to task B, and treat the two tasks independently. For task A, two methods of predicting topic behavior were tried: i) predicting the weakest and strongest n topics by SVM regression; and ii) ranking topics by retrieved document similarity. For task B, we followed the strategy introduced by us before to improve ad-hoc retrieval via the web as an external thesaurus to supplement a given topic’s representation, followed with data fusion. A new method of salient term selection from longer description queries based on SVM classification was employed to define web probes for these queries.

Five runs were submitted: two for title (pircRB05t2 and -t3), two for description queries (pircRB05d1 and –d3), and one for the combination of the two (pircRB05td3). Section 2 describes our experiments for topic prediction; section 3 describes our weak query effectiveness improvements; and section 4 has our conclusion.

2 Topic Rank Prediction

We have employed two methods for this task. One uses regression on a reduced problem of predicting only the strongest and weakest topics, and another attempts to do rank prediction based on similarity between retrieved documents. The first method does not rely on a retrieval, and we assume that the ACQUAINT and TREC2004 collections have similar (news-feed) properties, and that trained models from TREC2004 can be transferred to ACQUAINT. The respective methods and their results are discussed below.

2.1 Top-n and Bot-n Objectives for Prediction

A number of attempts were tried to predict the ranking of a set of topics before retrieval based on training and regression, particularly for the weakest topics. The end result was that we were rather pessimistic that such total order prediction can be accomplished with much accuracy. The reasons for a topic to return low effectiveness can be very many and varied. There are only a few hundred evaluated topics (and tens of weak ones) for training which is clearly insufficient.

Even if training data is increased substantially, there is still difficulty in identifying the right features that can differentiate which one topic should be ranked ahead of another, or discriminate weak topics from strong ones. We finally decided to reduce this prediction to (perhaps) an easier problem as discussed below.

The idea is that given a set of N topics, the n strongest (top-n) and the n weakest topics (bot-n) should have the largest differences in behavior, and hence features to characterize these differences may be easier to apply or identify. During training, we reduce the objective of predicting total ordering of topics to one of predicting only the identity of the top-n and bot-n topics, and ignoring all intermediate ones. We presume that what is learnt can then be applied to predict a rough ranking for all the topics. We define new objectives for training as follows. Suppose we have N=10 queries (labeled a to j), and the focus is on the n=3 worst and best ones. Assume that a system produces a topic ranking as indicated in the 'observed' row (the gold standard), and an algorithm predicts ranking results in the 'predicted' row as shown in Table 1:

bot(weakest) top(strongest)

rank->	1	2	3	4	5	6	7	8	9	10
observed->	a	b	c	d	e	f	g	h	i	j
predicted->	i	c	b	a	f	e	j	d	g	h

Table 1: 'Observed' and 'Predicted' Ranking of Topics

This example shows agreement in bot-3 is 2 (topics b, c), while that for top-3 is 1 (topic h). These are our primary objective; the larger these values are the better. In addition, one can see where the unmatched 'gold standard' topics go in the predicted list: e.g. topic a is ranked 4 by prediction (distance 1 from the bot-3 region), while topics i, j are ranked 1 and 7 (distances 7 and 1 respectively from top-3). An average observed distance 'obs.dis' can be used to characterize these mis-matches: 1/3 and (7+1)/3=2.67 respectively (see Table 2 below). The smaller these distances are the better, and they can be employed to break ties when the primary metric values are equal for two different predicted runs. Similarly, one can also ask where do the unmatched 'predicted' topics come from the observed list: e.g. topic i is ranked 9 by observation (distance 6 from bot-3) and topics d, g are ranked 4 and 7 by observation (distances 4 and 1 from top-3). An average predicted distance 'pre.dis' can also be calculated as 6/3=2 and (4+1)/3=1.67 respectively.

	bot-3	**pre.dis**	**obs.dis**	**top-3**	**pre.dis**	**obs.dis**
predicted	2(bc)	2	.33	1(h)	1.67	2.67
δ	2	1.67	0	1	.67	1.67
Perfect and Best Predictions with Agreements 2 and 1						
perfect	3	0	0	3	0	0
best-2	2	.33	.33	2	.33	.33
best-1	1	1	1	1	1	1

Table 2: Evaluation of 'Predicted' by top-3/bot-3 Agreements & Distances

In Table 2 are also shown the perfect prediction, and others that agree on only 2 or 1 in bot-3 and top-3 respectively, and their best 'distance' values. 'obs.dis' informs us where unmatched 'gold standard' topics have ended up in the predicted list, and 'pre.dis' informs us where the replacements come from the observed list. These can be compared with the 'best-n' distances, and displayed in the δ (difference from best) row to show how far these distances are from the 'best-n' values. One can focus only on the bot-3 with an evaluation measure for this prediction of <bot-3, δ> = <2,1.67>, or on the top-3 with measure <1, 2.34>. Clearly the bot-3 prediction is much better than top-3 for this example. One can also combine both bot-3 and top-3 together with measure <bot-3+top-3, δ> = <3, 4.01>. For this, the perfect prediction has measure <6,0>, and the best with 3 matches is <3,0>. Our objective is to maximize 'bot-3+top-3' and minimize δ.

2.2 Prediction by Regression

We continue to use the LIBSVM software (Chang & Lin 2004) to define a support vector regressor based on training and prediction using TREC2004 data with 249 topics. The features we employed include:

(a) $idf_k = \log(Nd/df_k)$ for the top three terms in a query. 'top' is defined later, and Nd = number of documents, df_k = document frequency of term k;

(b) 5-interval 'average term frequency' (v_k) distribution of all query terms. $v_k = (Cf_k/df_k)^{1.5}/[\log \max(\text{constant}, df_k)]$, defined in (Kwok 1996), and Cf_k = collection frequency of term k;

Here, idf_k (or its variants) is the basis of term weighting for many retrieval algorithms and is a natural choice. We choose the top three terms based on sorting by v_k. This seems to give better results than via sorting by idf_k. v_k has the (Cf_k/df_k) factor which is similar to term burstiness used in computational linguistics (Katz 1997) and can be useful for indicating whether a term has important content value. We modified it by a $1/[\log \max(\text{constant}, df_k)]$ factor in (Kwok 1996) that depresses the value for high df_k terms. The 5-interval v_k distribution of terms intends to capture the quality of the whole query through its term importance. The intervals are chosen as: $df_k < 2000$, $df_k > 40000$, $v_k < .24$, $v_k < .3$, $v_k >= .3$. The high and low thresholds are defined by frequencies instead, because some terms in these ranges may have abnormal high or low v_k values. Terms satisfying these thresholds sequentially have their v_k values summed for each of the intervals. These features have been shown to provide reasonable prediction of bot-n and top-n (Kwok 2005).

	Weak Topic Improvement			**Topic Ranking Prediction**	
RunID	**GMAP**	**MAP**	**P10**	**'diff-area'**	**Method**
Title					
pircRB05t2	.1957	.2798	.5420	1.4823	svm
pircRB05t3	.1905	.2769	.5380	1.9402	similarity
best	*.2326*	*.3323*	*.5920*	*0.0016*	
median	*.1293*	*.2239*	*.4340*	*1.4250*	
Description					
pircRB05d1	.1252	.2303	.4660	1.7325	similarity
pircRB05d3	.1208	.2228	.4640	1.2092	svm
best	*.1784*	*.2891*	*.5360*	*0.4049*	
median	*.1028*	*.1839*	*.3860*	*1.2262*	
Title+Description					
pircRB05td3	.2047	.2813	.5440	1.6065	copy from pircRB05d3
best	*.2326*	*.3323*	*.6280*	*0.0016*	
median	*.1256*	*.2182*	*.4320*	*1.4093*	

Table 3: Results of Title, Description, and Title+Description Submissions

2.3 Results of Prediction by Regression

pircRB05t2, pircRB05d3 are runs with the above bot-n/top-n prediction procedures. Their results are tabulated in Table 3 under the ‘Topic Ranking Prediction’ columns. They return ‘diff-area’ values of 1.4823 and 1.2092 respectively. The best and median values for title runs are .0016 and 1.4250, while those for description runs are .4049 and 1.2262. The best values may be biased by some runs that have low or close to zero average precision values for their topics. Prediction for our title run is below median; description prediction is above. For pircRB05td3, because of time constraint, the pircRB05d3 ranking was used for its prediction (‘Method’ svm-). Its ‘diff-area’ value of 1.6065 is below the median of 1.4093 for ‘other’ runs.

For the two SVM prediction runs, we also show in Table 4 predictions of bot-6 and top-6 values. The ‘-04’ attached to a Run-ID means using the TREC-2004 robust collection for the same procedures trained (with 199 topics) and testing on TREC-2005 robust topics. On average there were about 8/24=1/3 correct predictions. The ‘diff-area’ values are also good: about .7-.8. However, the trained regression model has problem generalizing to the ACQUAINT collection. Correct predictions now drop by half to an average of 4/24=1/6. Moreover, all the secondary observed and predicted ‘distance-difference-from-best’ measures increase also, except for top-6 ‘pre.dis’. Even if one stays with the TREC-2004 collection, i.e. without the complication of cross-collection generalization, the 1/3 accuracy rate is probably too low for any real application of this prediction process.

	bot-6	**pre.dis**	**obs.dis**	**top-6**	**pre.dis**	**obs.dis**	**‘diff-area’**
pircRB05t2-04	2	10.0	2.0	1	13.3	10.5	.7112
pircRB05d3-04	3	10.3	4.8	2	5.7	10.0	.7888
pircRB05t2	1	14.0	12.7	1	3.8	17.5	1.4823
pircRB05d3	0	16.7	10.7	2	3.0	11.5	1.2092

Table 4: Bot-6 and Top-6 Predictions for Title and Description Submissions

2.4 Prediction by Retrieved Document Similarity

A query q retrieves ordered documents d_i from a collection C. Each d_i can be represented by a vector of weights corresponding to each term. We assume these documents can characterize the query and the quality of the retrieval. The rationale for our approach is as follows: 1) in a ‘good’ retrieval the relevant documents are concentrated towards the top ranking positions; 2) relevant documents are more similar to each other than non-relevant documents; 3) a strong query retrieves a more cohesive group of documents; a weak query does not represent a well-defined concept and will retrieve many diverse, unrelated documents.

Our method consists of the following steps, which measures the similarity of the top group of retrieved documents to the remainder:

1) Divide the ranked retrieved documents into m groups, by rank.
2) Combine each group into one large document vector.
3) Measure the vector similarity of each group to the top group.
4) Combine the similarities by calculating the average, or some weighted formula, where higher ranked groups may be considered more important.

To test the algorithm we used the top 200 documents retrieved for each query by our PIRCS engine. m, the number of groups was set to 5 after experimenting with values between 4 and 20. Two common similarity measures were tried, vector product and covariance. No appreciable difference was detected and we selected the vector measure. The number of terms used for the groups was set to 40 after experimenting with values between 20 and 200. We tested whether to compare the top group to the others or to compare all groups to each other, with the result that comparing the top group seems to be a better predictor.

2.5 Results of Prediction by Retrieved Document Similarity

The runs that make use of this method are pircRB05t3 and pircRB05d1. The 'diff-area' values are 1.9402 and 1.7325 which are disappointing. Their correlation between the predicted query rank and the actual rank was negligible. We are currently reviewing the results to determine whether the problem lies in our algorithm or the experimental procedure, or perhaps the distributions of retrieval results from the two collections are very different.

3 Improving Ad Hoc Retrieval

Our results for this year's robust tasks are also summarized in Table 3. GMAP is the geometric mean average precision specially designed to evaluate low performing topics with heavier emphasis. The method is similar to last year's: construct web probes from a given query, use the web as an all-domain thesaurus to define alternate queries, combine retrieval lists from original and alternate queries after retrieval from ACQUAINT collection. The sub-sections below discussed our procedures for the different topic types.

3.1 Title Queries

Title (1 to 4 words long) or title words and phrases (formed via MINIAPR parsing (Lin 1994)) defined the web probes. Three alternate queries were formed from web output of full pages, snippets or both. Retrieval lists from ACQUAINT collection were combined two by two with coefficients trained via TREC2004 collection. This is our pircRB05t2 submission. pircRB05t3 makes use of only one alternate query that combines with the original title retrieval list. We like to see how results from this much simpler procedure differ from the more elaborate one of combining four lists.

Results show that pircRB05t2 attains GMAP, MAP and P10 values of 0.1957, 0.2798 and 0.5420 respectively, while the corresponding values for pircRB05t3 are 0.1905, 0.2769 and 0.5380. pircRB05t3 provides slightly weaker effectiveness than the more elaborate pircRB05t2. However, the decrease for GMAP is nearly 3%, more than the 1% decrease for the other traditional metrics. Combining more different lists can provide opportunity for more weak queries to get improved (Kwok et. al. 2005).

3.2 Description Queries

To select salient terms from a description query as web probe, we introduced term classification based on a SVM model trained via the paired title words as positive samples (Kwok et. al. 2005). The selected terms are used in two ways: i) to enhance the original query by double weighting the selected terms (enhanced description query); ii) used for web probing and form alternate queries. In addition, two other alternate queries were formed via term selection by MINIPAR-tagged POS. pircRB05d1 combines 4 lists and pircRB05d3 combines 2 lists, similar to title runs.

Again, pircRB05d1 gives better GMAP results of .1252 vs. .1205 for pircRB05d3, a decrease of nearly 4%. MAP and P10 measures decrease less similar to the behavior of title queries.

3.3 Title + Description Queries

Our best submitted result is obtained by combining the pircRB05t3 and pircRB05d3, which involves combining four lists, two from title and two from description runs. In terms of combination complexity, it is the same as a pricRB05t2 or –d1 runs. It's GMAP, MAP and P10 values are .2047, .2813 and .5440 respectively.

3.4 Collection Effect on Topic Strength

In our experiments, the same queries were derived and trained from the TREC2004 collection and transferred to ACQUAINT for retrieval. Thus, we have comparable results from the two collections, and their effects on how queries behave can be ascertained. Fig.1a shows results of retrieval from the two collections sorted according to the weakness of title queries (pircRB05t2), and Fig.1b for the title+description run (pircRB05td3). It is seen that in both plots, results from TREC2004 are inferior to those from ACQUAINT except for the one weakest query in Fig.1b. This accounts for the difference of GMAP values for pircRB05t2: .1958 vs. .1308 (TREC2004) and for pircRB05td3: .2047 vs. .1456 (TREC2004). Some eleven individual queries perform worse in ACQUAINT than in TREC2004 with decrease of average precision of .1180, while thirty nine improve with average precision of .1432. Topic difficulty is thus a relative concept: the same query may be weak for collection TREC2004 but reasonable or even strong for collection ACQUAINT. If one makes use of multiple external collections as aids for target retrieval (e.g. collection enrichment (Kwok & Chan 1998)), the probability of improving weak query effectiveness should be higher. This indirectly also shows that our procedure of appealing to the web as an all-domain thesaurus to help target retrieval via alternate queries is a step in the right direction for the objective of the robust track.

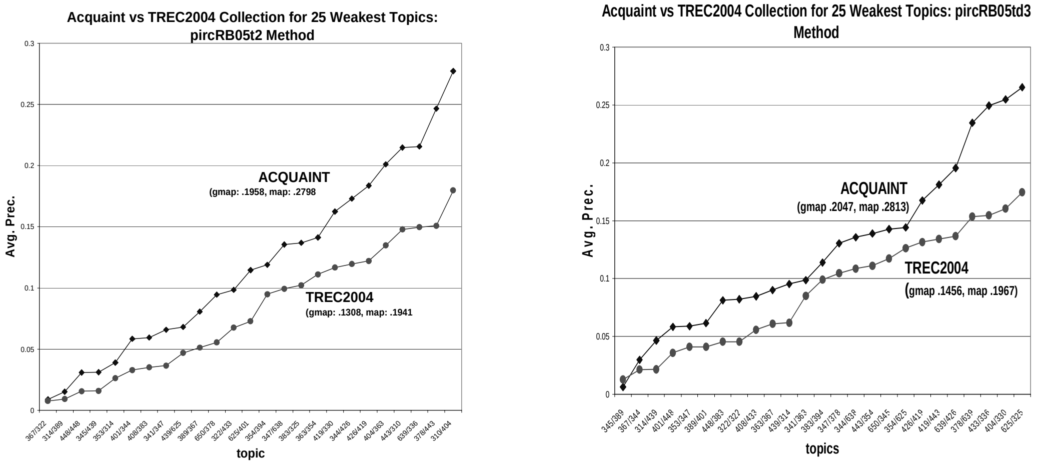

Fig.1: Effect of Collections on Query Weakness. a) pircRB05t2; b) pircRB05td3

4 Conclusion

We investigated an approach of training an SVM-based regression model for predicting query performance by trying to predict only the best (top-6) and the worst (bot-6) topics from an initial retrieval in the TREC-2004 collection, and allowing the model to forecast the performance of the same topic set in the ACQUAINT collection. The result of this cross-collection prediction is much worse than predictions within the TREC-2004 collection.

For the improvement of query performance, our procedure of defining alternate queries via web-assistance and later combining their retrieval lists with the original one continue to deliver above average results.

References

Chang, C-C and Lin C-J (2004). LIBSVM: a Library for Support Vector Machines. Available at: http://www.csie.ntu.edu.tw/~cjlin/libsvm.

Katz, S (1996). Distribution of content words and phrases in text and language modeling. Natural Language Engineering, 2(1), pp.15-59.

Kwok, K.L (2005). An attempt to identify weakest and strongest queries. In Proc. of SIGIR 2005 Workshop on Predicting Query Difficulty. 2005. (http://www.haifa.il.ibm.com/sigir05-qp/papers/kwok.pdf)

Kwok, Kui-Lam, Grunfeld, Laszlo & Deng, Peter (2005). Improving Weak Ad-Hoc Retrieval by Web Assistance and Data Fusion. In: Proc. of Asian Information Retrieval Symposium 2005. Springer-Verlag LNCS 3689. pp.17-30.

Kwok, K.L & Chan, M. (1998) Improving two-stage ad-hoc retrieval for short queries. Proc. 21st Ann. Intl. ACM SIGIR Conf. on R&D in IR. pp.250-256.

Kwok, K.L. (1996) A new method of weighting query terms for ad-hoc retrieval. Proc. 19th Annual Intl. ACM SIGIR Conf. on R&D in IR. pp.187-195.

Lin, D. (1994). PRINCIPAR – an efficient, broad-coverage, principle-based parser. *Proc of COLING-94*. pp.482-488

Voorhees, E.M. (2005). Overview of the TREC 2004 Robust Retrieval Track. In: Information Technology: The Thirteenth Text REtrieval Conference, TREC 2004. E.M. Voorhees & L.P. Buckland, Editors. NIST Special Publication 500-261, US GPO: Washington, DC. pp.70-79.