Experiments with Positive, Negative and Topical Relevance Feedback

Rianne Kaptein[1] **Jaap Kamps**[1,2] **Rongmei LI**[3] **Djoerd Hiemstra**[3]

[1] Archives and Information Studies, Faculty of Humanities, University of Amsterdam
[2] ISLA, Informatics Institute, University of Amsterdam
[3] Database Group, University of Twente

Abstract This document contains a description of experiments for the 2008 Relevance Feedback track. We experiment with different amounts of feedback, including negative relevance feedback. Feedback is implemented using massive weighted query expansion. Parsimonious query expansion using Dirichlet smoothing performs best on this relevance feedback track dataset. Additional blind feedback gives better results, except when the blind feedback set is of the same size as the explicit feedback set. On a number of topics topical feedback is applied, which turns out to be mainly beneficial for early precision.

1 Introduction

In this first year of the Relevance Feedback track we experiment with several relevance feedback approaches. Evaluation of feedback approaches is complicated because interaction with the system is dynamic, and performance depends on the feedback of users. Standard TREC evaluation measures are static and do not have a natural way to incorporate feedback [5]. The Relevance Feedback track is a first attempt to set up a framework in which relevance feedback approaches can be studied, evaluated and compared.

This track allows us to explore the effects of using different amounts of relevance feedback, positive as well as negative feedback. In addition, we experiment with another form of feedback, namely topical feedback. Instead of using relevant documents, topical feedback uses topic categories considered relevant to the query. To cope with the dynamic nature of the task, all feedback documents are removed from the result ranking before evaluation, creating a so-called residual ranking, on which the standard evaluation measures can be applied. Another option would be to freeze the feedback documents on their position in the initial ranking [1].

The rest of this paper is organized as follows. In Section 2, we discuss the details of the models we use for relevance and topical feedback. In Section 3, we first describe the experimental set-up, and then our experiments on the training and test data. Finally, we draw our conclusions in Section 4.

2 Models

We use different models in order to incorporate feedback from positive and negative relevance feedback and topical feedback.

2.1 Relevance Feedback

Relevance feedback is applied using an adaptation of Lavrenko and Croft's Relevance Model [3]. Their relevance model provides a formal method to determine the probability $P(w|R)$ of observing a word w in the documents relevant to a particular query. The method is a massive query expansion technique where the original query is completely replaced with a distribution over the entire vocabulary of the relevant feedback documents. Instead of completely replacing the original query, we include the original query with a weight W_{orig} in the expanded query.

For all our experiments we use the Indri search engine [6]. Our baseline model is a standard language model. In the original baseline query Q_{orig} each query term gets an equal weight of $\frac{1}{|Q|}$.

Our first relevance feedback approach only uses positive relevance feedback. The approach is similar to the implementation of pseudo-relevance feedback in Indri, and takes the following steps:

1. $P(t|R)$ is estimated using the given relevant documents either using maximum likelihood estimation, or using a parsimonious model [2].

   The parsimonious model is estimated using *Expectation-Maximization*:

   $$\text{E-step:} \quad e_t = tf(t,R) \cdot \frac{(1-\lambda)P(t|R)}{(1-\lambda)P(t|R) + \lambda P(t|C)}$$

   $$\text{M-step:} \quad P(t|R) = \frac{e_t}{\sum_t e_t}, \text{i.e. normalize the model}$$

   In the M-step terms that receive a probability below a threshold of 0.001 are removed from the model. In the next iteration the probabilities of the remaining terms are again normalized. λ determines the weight of the background model $P(t|C)$.

2. Terms $P(t|R)$ are sorted, in case of MLE only the 50 top ranked terms are kept.

3. The relevance feedback part, Q_R, of the expanded query is constructed as:

$$\#weight(P(t_i|R)\ t_i\ ...\ P(t_n|R)\ t_n)$$

4. The fully expanded Indri query is now constructed as:

$$\#weight(W_{orig}\ Q_{orig}\ (1 - W_{orig})\ Q_R)$$

5. Documents are retrieved based on the expanded query

2.2 Negative Feedback

Until now, we only used the relevant feedback documents. Most of the feedback document sets also contain non-relevant documents. We experiment with two approaches to also take into account the non-relevant feedback documents. For both approaches we first estimate a parsimonious model for the relevant documents $P(t|R)$ and a parsimonious model for the negative documents $P(t|N)$. Typically some words, including the query terms, will occur in both the negative and the positive documents.

The first approach (Comb QE) divides all terms in the positive model by their value in the negative model, or by a factor α if the term does not occur in the negative model. The probabilities are afterwards normalized to add up to 1. For α we use the value 0.001, which is equal to the threshold used in the parsimonious model estimation. This approach boosts probabilities of terms occurring in the positive but not in the negative model, assuming these terms will make a better distinction between relevant and non relevant documents.

The second approach (Neg QE) takes the positive model and adds all terms from the negative model that do not occur in the positive model with a negative weight. This approach is based on the assumption that if a term occurs in both the positive and the negative model, it is still a good term to use for feedback.

Both models are extensions to the original query, where the original query has a total weight of 1.

2.3 Topical Feedback

Besides the given relevance feedback sets, there are also some manual topics for which participants in the track can define their own relevance feedback. In our case we use topical categories as topical feedback. A topical category from the DMOZ directory is assigned to each query. We assume that all web sites in the chosen DMOZ category, and all of its direct subcategories are relevant to the query. The topical feedback model is build from the text on these web sites. Topical feedback is applied in the same way as explicit relevance feedback where instead of the relevant document(s) $P(t|R)$, we now have the topical model $P(t|TM)$.

We implemented a second variant of the topical model, where the weights of the original query are adjusted according to the fraction of query words in the topical category title. If the query terms are equal to the category title, this topical model is a good match for the query, so the weight of the topical model terms can be high. On the other hand, if none of the query terms occur in the category title, it is unlikely that the topical feedback will contribute to retrieval performance, so the weight of the topical feedback is lowered. The original weights of the query words are $\frac{1}{|Q|}$, the adjusted weights of the querywords are $1/(|Q| * \text{fraction of query terms in category title})$. A fraction of $1/5$ is used when none of the query terms occur in the category title.

3 Experiments

3.1 Experimental Set-up

The Relevance Feedback track test topics consist of 50 (even-numbered) topics from the Terabyte tracks and 214 (even-numbered) topics from the 2007 MQ track. We train on the odd-numbered Terabyte topics, since for these topics extensive relevance judgments are available.

For efficiency reasons we do not build an index of the complete .GOV2 collection. Instead we build an index using only the top 2,500 results of runs that we made in previous Terabyte and Million Query tracks. These previous runs are created by using a standard language model, with Jelinek-Mercer smoothing (λ = 0.1). We build one index which contains both the training and the test data. This index contains 742,664 documents, 9,228,163 unique terms and a total of 4,860,799,852 terms. Since this background corpus is much smaller, contains longer documents, and is biased towards the queries, the estimations of background probabilities may not reflect the whole corpus well.

For the training data no relevance feedback document sets are given, so we create these by taking the highest ranked documents of our Terabyte track run. The feedback sets contain the following documents:

- Set B: 1 relevant document
- Set C: 3 relevant, and 3 non-relevant documents
- Set D: 10 documents, set C always included
- Set E: All previously judged documents (for training only 100 documents)

3.2 Baseline

We use the language model of Indri for our experiments. To incorporate the explicit relevance feedback, we use weighted query expansion.

Besides the explicit relevance feedback we also do blind relevance feedback, based on Lavrenko and Croft's relevance model. Indri's blind relevance feedback is applied using parameters from [4], i.e., number of feedback documents = 10, terms for query expansion = 50, weight original query = 0.5, μ= 1500. In addition we also use our own scripts to apply blind relevance feedback using query expansion in the

Table 1: Baseline results

Smoothing	Blind FB	Prior	MAP	Bpref	P10
JM	No	No	0.2135	0.2930	0.3595
JM	Indri	No	0.2645	0.3343	0.4500
Dir.	No	No	0.2837	0.3341	0.5446
Dir.	No	Yes	0.2774	0.3323	0.5500
Dir.	Indri	No	**0.3155**	0.3618	**0.5797**
Dir.	QE	No	0.3021	**0.3727**	0.5500

same way as our explicit feedback. Again we use the top 10 retrieved documents.

We have made a number of baseline runs, that do not use explicit relevance feedback. The results on the training data, i.e. 75 odd-numbered Terabyte Track queries, are given in Table 1. The following parameters can be adjusted:

- Two smoothing techniques are used: JM stands for Jelinek-Mercer smoothing with $\lambda = 0.1$, Dir. stands for Dirichlet smoothing with $\mu = 1500$.
- A document prior based on document length (length prior).
- Blind relevance feedback, either using indri with the parameters given above (Indri), or by using query expansion (QE).

On our baseline runs Dirichlet smoothing achieves significantly better results than Jelinek-Mercer smoothing. Indri's blind feedback performs better, except on Bpref, than doing query expansion with our own scripts, probably due to a better optimization of parameters. From now on, when we apply blind feedback, we use Indri's blind feedback. Applying the length prior leads to a decrease in MAP and Bpref, but to an increase in P10. We will not apply a length prior in any of the other runs.

3.3 Relevance Feedback

Table 2 gives the results of applying relevance feedback using one relevant document as feedback (set B). Relevance feedback documents are used for query expansion, either using Maximum Likelihood Estimation (MLE QE) or a parsimonious model (Pars QE). In case of Maximum Likelihood Estimation the top 50 terms are used, and their probabilities are normalized to add up to 1. The parsimonious model uses a λ of 0.01, and a threshold of 0.001. The original query terms are included in the query with a total weight of 1, the weight of the added query terms together is also 1, which is the same as using $W_{orig} = 0.5$.

Our purpose here is to find the optimal parameters for this feedback set. Therefore, in this section before evaluation we only remove the given relevant document or documents from the ranking. Although it becomes more difficult to compare across different feedback sets, results within one feedback set are more accurate.

Table 2: Results feedback set B

QE	Smoothing	Blind FB	MAP	Bpref	P10
None	Dir.	Yes	0.3044	0.3531	0.5500
Pars	JM	No	0.3205	0.3873	0.5662
MLE	JM	No	0.3055	0.3774	0.5608
Pars	Dir.	No	0.3198	0.3737	0.6216
MLE	Dir.	No	0.3152	0.3728	0.6189
Pars	JM	Yes	0.3239	**0.4066**	0.5892
MLE	JM	Yes	0.3199	0.4007	0.5865
Pars	Dir.	Yes	**0.3300**	0.3919	**0.6405**
MLE	Dir.	Yes	0.3266	0.3920	0.6338

Comparing parsimonious and MLE query expansion, parsimonious query expansion consistently gives slightly better results, but the improvements are very small and not in all cases significant. For the other feedback sets we will always use parsimonious query expansion. The differences between Dirichlet and Jelinek-Mercer smoothing are much smaller here, only P10 seems to be better when Dirichlet smoothing is used. These results adhere to the results of the comparison of smoothing techniques in [7]. They find Dirichlet smoothing performs best on short queries, i.e. no query expansion. For long queries, i.e. when query expansion is used, Jelinek-Mercer is on average better, but average precision is almost identical to Dirichlet smoothing. For feedback set B, applying blind feedback on top of the explicit relevance feedback leads to considerable improvements.

Tables 3 to 5 give the results using feedback sets C, D and E. For these sets also non-relevant documents are provided. We use this negative feedback in two ways. The first method (Comb QE) divides all terms in the positive feedback model by their value in the negative model. The second method (Neg QE) takes the positive model and adds all terms from the negative model that do not occur in the positive model with a negative weight. For the feedback sets C, D and E we also still do query expansion using only the positive feedback documents. Results of the different query expansion methods depend also on the smoothing technique that is used.

Using feedback set C results of the three query expansion methods lie very close together, and there is not one method that is best for all evaluation measures. The combination of Jelinek-Mercer smoothing and combined query expansion gives the best MAP and Bpref. Best P10 is achieved using parsimonious query expansion and Dirichlet smoothing.

For feedback set D, parsimonious query expansion is best on all three evaluation measures. On the training data, using the negative relevance feedback information does not lead to better results than only using positive relevance feedback. Comparing the two methods (Comb QE and Neg QE), differences are small, combined query expansion in combination with Jelinek-Mercer smoothing seems to be the most promising approach.

Looking at all results, in general Dirichlet smoothing is

Table 3: Results feedback set C

QE	Smoothing	Blind FB	MAP	Bpref	P10
None	Dir.	Yes	0.2965	0.3468	0.5622
Pars	JM	No	0.3261	0.3869	0.5946
Pars	JM	Yes	**0.3353**	**0.4095**	0.6230
Pars	Dir.	No	0.3291	0.3794	**0.6473**
Pars	Dir.	Yes	0.3341	0.3945	0.6405
Comb	JM	No	0.3298	0.3934	0.6257
Comb	Dir.	No	0.3247	0.3772	0.6446
Neg	JM	No	0.2967	0.3691	0.5554
Neg	Dir.	No	0.3243	0.3823	0.6311

Table 4: Results feedback set D

QE	Smoothing	Blind FB	MAP	Bpref	P10
None	Dir.	Yes	0.2741	0.3299	0.5405
Pars	JM	No	0.3082	0.3761	0.5770
Pars	Dir.	No	**0.3123**	0.3701	**0.6365**
Pars	Dir.	Yes	0.3110	**0.3810**	0.6216
Comb	Dir.	No	0.3081	0.3678	0.6243
Neg	Dir.	No	0.3083	0.3767	0.6297

Table 5: Results feedback set E

QE	Smoothing	Blind FB	MAP	Bpref	P10
None	Dir.	Yes	0.1079	0.2088	0.3176
Pars	JM	No	0.1341	**0.2517**	0.3946
Pars	Dir.	No	0.1343	0.2431	0.4108
Pars	Dir.	Yes	**0.1394**	0.2504	**0.4365**

to be preferred. Differences in MAP and Bpref are small, and sometimes Jelinek-Mercer smoothing also gives better results. Dirichlet smoothing however does give consistently better P10 values.

While for feedback sets B and C applying additional blind feedback still leads to an increase in improvement, for feedback sets D and E there are no real improvements. The explicit feedback sets D and E are equal or larger than the set of documents used for blind relevance feedback. Since the feedback sets B to E are selected using an initial run very similar to the our new run, there will be a large overlap in the explicit feedback and the blind relevance feedback documents for the top 10 ranked documents. Feedback set D consist of the top 10 documents and is therefore the most similar to the blind feedback set of the top 10 ranked documents. For feedback set D, we see that applying additional blind relevance feedback leads to a decrease in MAP and P10, but an increase in Bpref. For feedback set E applying blind feedback leads to a small increase in performance on all three measures. Feedback set E contains of the first 100 documents, of which in this case only the relevant documents are used. Using this large amount of documents possibly leads to less focused query expansion terms, which can be corrected partly by including blind feedback using only the top 10 ranked documents.

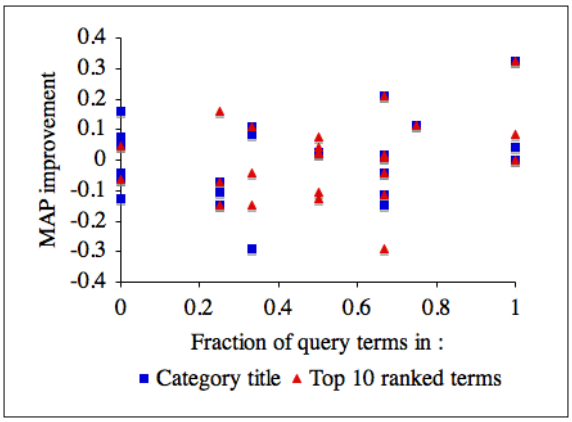

Figure 1: MAP improvement correlations

Table 6: Results manual topics

QE	Blind FB	Prior	MAP	Bpref	P10
None	No	No	0.2902	0.3415	0.5680
None	Yes	No	0.3267	0.3736	0.6120
Topic	No	No	0.2694	0.3392	0.5560
Topic	No	Yes	0.2789	0.3541	0.5160
Topic	Yes	No	0.3069	0.3710	0.5760
W. Topic	No	Yes	0.3023	0.3616	0.5560
W. Topic	Yes	Yes	**0.3339**	**0.3847**	**0.6360**

3.4 Topical Feedback

We apply topical feedback on the manual topics of the RF track. For Terabyte topics 800–850 we use topical categories assigned by test users in a user study. For the other topics topical categories are assigned by ourselves. We use odd-numbered topics 800–850 from the Terabyte track for training. Besides the standard topical query expansion (Topic QE), we also give results of the weighted topical query expansion (W. Topic QE). To create the topical model we use a λ of 0.01, and a threshold of 0.001. In each run we use Dirichlet smoothing. The parameters are whether blind feedback is applied, and whether a document length prior is used.

The weighted topical query expansion works because there is a weak (non-significant) correlation between improvement in MAP when topical query expansion is used, and the fraction of query terms in either the category title, or the top ranked terms of the topical language model, as can be seen in Figure 1.

Results of the manual topic runs can be found in Table 6. Although on average the topical model feedback only leads to a small improvement of MAP over the baseline, for 8 out of 25 topics, the topical model feedback has best MAP of all models. In the run Weighted Topic QE, we reweigh the original query terms according to the inverse fraction of query terms that occur in the category title, i.e. if half of the query terms occur in the category title, we double the original query weights. These runs lead to better results and to improvements over blind relevance feedback, but they are

Table 7: Results test runs

Set	QE	Smoothing	MAP	Bpref	P10
A	None	Dir.	0.1574	0.2296	0.2871
A	None	JM	0.1222	0.2205	0.2258
B	Pars	Dir.	0.1930	0.2642	0.3516
B	Pars	JM	0.2017	0.2792	0.3903
B	Comb	Dir.	0.1930	0.2642	0.3516
B	Comb	JM	0.2017	0.2792	0.3903
C	Pars	Dir.	0.1989	0.2713	0.3774
C	Pars	JM	0.2116	0.2869	0.3968
C	Comb	Dir.	0.1898	0.2665	0.3871
C	Comb	JM	0.1895	0.2663	0.3903
D	Pars	Dir.	0.2059	0.2867	0.3484
D	Pars	JM	0.2120	0.2927	0.3806
D	Comb	Dir.	0.2000	0.2846	0.3742
D	Comb	JM	0.1898	0.2781	0.3774
E	Pars	Dir.	0.2058	0.2909	0.3839
E	Pars	JM	**0.2139**	0.2985	0.3806
E	Comb	Dir.	0.2132	0.2940	**0.4226**
E	Comb	JM	0.2131	**0.3037**	0.4161

not significant on our small training set of 25 topics.

3.5 Test Results

On the test data we experiment with smoothing and query expansion methods. We make four runs using either Dirichlet or Jelinek-Mercer smoothing, and either parsimonious or combined query expansion. All runs apply additional blind relevance feedback. The test data consist of 31 Terabyte track topics that are evaluated approximately according to the standard TREC evaluation strategy. All documents from feedback set E are removed before evaluation takes place.

The results are given in Table 7. Considering smoothing techniques, the results are similar to the training results, i.e. there is little difference between results, but in most cases Dirichlet smoothing leads to better results, especially on early precision. Comparing parsimonious query expansion with combined query expansion, there is no clear winner. Combined query expansion leads to better early precision, but on the average precision measures there are no notable differences. When we look at the different feedback sets, we notice that more relevance information does not always lead to better results. The biggest improvements by far are achieved when going from no relevance feedback to using one relevant document. Part of this improvement might be attributed to the smoothing parameter settings, which are optimized for long queries.

Table 8 shows the results for topical feedback. It does not lead to significant improvements over the baseline on the 13 test topics, for MAP no improvement at all is achieved. We do achieve more than 8% improvement in P10.

4 Conclusions and Future Work

From our experiments with different relevance feedback approaches we can conclude that our query expansion approach is effective, already with small amounts of relevance information. There are no significant differences between the different smoothing and query expansion approaches. Additional blind feedback gives better results, except when the blind feedback set size is equal to the relevance feedback set size.

Table 8: Results manual topics test runs

QE	Prior	MAP	Bpref	P10
None	No	**0.3873**	0.4416	0.6385
Topic	No	0.3412	0.4139	0.6615
Topic	Yes	0.3332	0.4212	**0.6923**
W. Topic	No	0.3811	0.4417	0.6615
W. Topic	Yes	0.3674	**0.4443**	0.6692

Topical feedback can be used as an alternative to relevance feedback. Improvements over blind relevance feedback are achieved, especially for early precision. We would like to explore in more detail the topical feedback approach, and how topical feedback relates to relevance feedback. We found some indicators to predict the performance of topical feedback on individual queries, and it would be interesting to continue investigating performance indicators.

In our experiments we have used an index that does not include the complete .GOV2 collection, but a subset of documents based on previous runs. Since the feedback approaches introduce new query terms in the expanded queries, we might retrieve new relevant documents, that are currently not in the index, when we index the whole collection.

Acknowledgments This research is funded by the Netherlands Organization for Scientific Research (NWO, grant # 612.066.513).

REFERENCES

[1] Y. K. Chang, C. Cirillo, and J. Razon. Evaluation of feedback retrieval using modified freezing, residual collection and test and control groups. In G. Salton, editor, *The SMART retrieval system – experiments in automatic document processing*, pages 355–370, 1971.

[2] D. Hiemstra, S. Robertson, and H. Zaragoza. Parsimonious language models for information retrieval. In *Proceedings SIGIR 2007*, pages 178–185. ACM Press, New York NY, 2004.

[3] V. Lavrenko and W. B. Croft. Relevance-based language models. In *Proceedings SIGIR 2001*, 2001.

[4] D. Metzler, T. Strohman, Y. Zhou, and W. B. Croft. Indri at trec 2005: Terabyte track. In *TREC: Experiment and Evaluation in Information Retrieval*, 2005.

[5] S. Robertson, S. Walker, and M. Beaulieu. Experimentation as a way of life: Okapi at TREC. *Information Processing & Management*, 36:95–108, 2000.

[6] T. Strohman, D. Metzler, H. Turtle, and W. B. Croft. Indri: a language-model based search engine for complex queries. In *Proceedings of the International Conference on Intelligent Analysis*, 2005.

[7] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings SIGIR 2001*, pages 49–56. ACM Press, 2001.