DUTIR at TREC 2011: Chemical IR Track

Ping Zhang[1,1], Hongfei Lin[1], Jiajin Wu[1], Yuan Lin[1]

[1]Information Retrieval Laboratory, Dalian University of Technology,
Dalian 116024
{pingzhang,wujiajin,yuanlin}@mail.dlut.edu.cn
hflin@dlut.edu.cn

Abstract. This paper presents the DUTIR submission to TREC 2011 Chemical IR Track. Several experiments are done mainly with two retrieval models i.e. Language Model for IR and DFR model. In addition, query expansion technology is also applied to enhance retrieval performance.

1 Introduction

In this paper, we describe our methods for the TS task and PA task. As only 4 runs are allowed to submit, we use Query likelihood model, DFR model, Phrase Expansion method and D-smoothing method to finish the TS task. But in the PA task, for at most ten runs could be submitted, we could try more methods for query generation and document retrieval, such as term frequency, tf/idf weighting, DFR model and re-rank mechanism.

2 Our methods for Technology Survey task

There are several fields could be used in the TS topics, such as title, narrative, chemicals and so on. In general, title field is the most important part in the topic. So we use title field, and adapt some query formatting methods to generate our query and use different retrieval model to get the results.

2.1 TSRun1-QL+PE

Our baseline run is *DUT11TSRun1*, in which we use **Q**uery-**L**ikelihood model as our retrieval model, and the Lemur4.12 toolkit [1] is used as our basic retrieval system. With the indri query system in Lemur, documents are retrieved for the given

This work is supported by grant from the Natural Science Foundation of China (No.60673039and 60973068), the National High Tech Research and Development Plan of China (No.2006AA01Z151), National Social Science Foundation of China (No.08BTQ025), the Project Sponsored by the Scientific Research Foundation for the Returned Overseas Chinese Scholars, State Education Ministry and The Research Fund for the Doctoral Program of Higher Education (No.20090041110002).

query by query-likelihood language model [2], in which D-smoothing method [3] is used. In this run, we empirically set the Dirichlet prior to 1500.

In addition, we also adapt the **P**hrase **E**xpansion model [4] during the process of query generation. The Phrase expansion model can be simply described as follows: query terms are likely to appear in close proximity to each other within relevant documents. Take the topic TS-28, "D-ala-D-ala ligase inhibitors" for example; relevant documents may contain the phrases "D-ala-D-ala" and "ligase inhibitors" within relatively close proximity to each other. The phrase expansion model takes this situation into consideration. For example, to the topic TS-20, we generate the final query:

#*weight* (0.85 #*combine* (*tests HCG hormone*)
0.1 #*combine* (#1(*tests HCG*) #1(*HCG hormone*) #1(*tests HCG hormone*))
0.05 #*combine* (#*uw8* (*tests HCG*) #*uw8* (*tests hormone*) #*uw8* (*HCG hormone*) #*uw12* (*tests HCG hormone*)))

With these methods, we generate our baseline run.

2.2 TSRun2-QL+PE+QE

DUT11TSRun2, which is based on the *DUT11TSRun1*, also employs the **Q**uery-**L**ikelihood retrieval model and use the **P**hrase **E**xpansion method to construct object query. Moreover, **Q**uery **E**xpansion technology is also employed in this run. We mainly use the pseudo-relevance Feedback (PRF) method, which has been shown to be an effective way of improving retrieval performance. We select the top 10 documents in the retrieved document list for a topic, and construct a relevance model from them. Then the original query is extended with the 50 terms that are not stopwords with the highest likelihood from the relevance model. The final form of the Indri query is: # (*weight* 0.8 #*combine* ($q_1 \ldots q_N$) 0.2 #*combine* ($e_1 \ldots e_{50}$))

2.3 TSRun3-In_expB2

The *DUT11TSRun3* is different from the first two runs, and the In_expB2 weighting model [5] is employed. This retrieval model is derived from the Divergence from Randomness (DFR) framework. The relevance score between a document d and a query q can be denoted as follows:

$$\begin{aligned} &\text{score}(d,q) = \sum_{t \in q} TF * qtf * norm * \log_e\left(\frac{N+1}{n\_\exp}\right) \\ &TF = tf * \log_2(1 + avdl / dl) \\ &norm = (tf + 1) / (df * (TF + 1)) \\ &n\_\exp = df * (1 - e^{-qtf/df}) \end{aligned} \tag{1}$$

where N denotes the number of the documents in the collection, tf is within-document term frequency, qtf is within-query term frequency, df is the number of documents in which term t is appeared, dl is the length of the document and $avdl$ is the average document length over the whole collection. It is a parameter-free model, so there is no

parameter that needs to be tuned. In addition, we use description field in the patent collection as the search field, and take title field in the given topic as the query. Then this run is generated.

2.4 TSRun4-In_expB2+QE

Our official run *DUT11TSRun4*, which is based on the *DUT11TSRun3*, is also used the DFR framework. Besides, **Q**uery **E**xpansion technology is adopted in this run. We set the description field as the expansion field, and we also select 10 documents in the first retrieval results as the expansion source. Then we select the top 50 terms with the highest likelihood as the expansion terms.

3 Our methods for Prior Art task

The Prior Art task contains a subtask that asks the participants to submit the results for the first 100 topics (PA-1001 to PA-1100) in the query patent list. We only submit the results for this subtask. With the combination of query generation methods and retrieval methods, we generate 9 runs in all. We divide them into four groups: Simple QL (*PARun1-3*)&Simple DFR (*PARun6*, *PARun8*), Query Generation & DFR (*PARun7*) and Query Generation & QL (*PARun4-5*, *PARun9*). We'll introduce them in details.

3.1 Simple QL & Simple DFR

This group of runs is similar with *TSRun1* and *TSRun2*. D*UT11PARun1*is a simply Query-likelihood run, and only uses title field in the topic as the query.*DUT11PARun2* is based on *DUT11PARun1*, and adopts the Phrase Expansion technology to form the object query. The *DUT11PARun3* is similar to *DUT11PARun2*; what's more, it also uses the query expansion technology. Simple DFR only takes words from title field in the topic as the query. *DUT11PARun6* uses In_expB2 weighting model as retrieval model. And *DUT11PARun8* is based on *DUT11PARun6*; in addition, query expansion technology described is applied.

3.2 Query Generation & DFR

The query patent contains at least four fields i.e. title field, abstract field, claims field and description field. In *DUT11PARun7*, we generate the query using the words from title field, description field and claims field. We calculate the *tf/idf* score of each term that is not stopword and appeared in the above three fields in a topic, and select the top 30 terms that appeared in the title field or claims filed with the highest *tf/idf* scores as the query. The *tf/idf* equation is denoted as follows:

$$\begin{aligned} Score_{Field_i}(e) &= \log(tf_i(e) * idf \;/\; fieldLength_i) \\ Score(e) &= \sum_i Score_{Field_i}(e) \end{aligned} \tag{2}$$

We also use In_expB2 weighting model as our retrieval model. These generate our official run *DUT11PARun7*.

3.3 Query Generation & QL

This group of runs uses the query-likelihood model as retrieval model and date filtering method to filter out the patents that are published after the query patent. As we all know, title field is important in the query patent. The *DUT11PARun4* takes the title field, description field and claims field into consideration. We use all title words and select sets of terms from the above three fields to construct our queries. First, we calculate the term frequency of each term appeared in the above fields; second, we select top 30 terms that appeared in title field or claims field with highest term frequency. The final form of the query is:

$$\#weight(w1\ \#combine(title)\ w2\ \#weight(tf_1\ term_1,\ tf_2\ term_2 \ldots\ tf_{30}\ term_{30}))$$

we empirically set parameter *w1* to 0.8 and *w2* to 0.2, where tf_i denotes the term frequency of $term_i$.

The *DUT11PARun5* is similar to the *DUT11PARun4* and the only different lies in that it takes the *tf/idf* score as the criteria of terms selection. With comparison to the DUT11PARun7, the difference is the selection of retrieval model.

Our final run *DUT11PARun9* is a re-rank process of *DUT11PARun5*. The citation network [6] is introduced. For each retrieved patent *i*, we use all the scores of patents that reference to patent *i* and appeared in the retrieval results to enhance patent *i*. The equation is denoted as follows:

$$FinalScore(i) = Score(i) + \sum_j is\_referenced_{i,j} * \alpha * Score(j) \quad (3)$$

where $Score(i)$ is the origin score generated by the retrieval system. The parameter $is\_referenced_{i,j}$ denotes that whether patent *j* is referenced to patent *i*. The parameter α is empirically set to 0.005.With this equation, we calculate the *FinalScore* for each patent in the retrieved results and sort the patents by *FinalScore* in descending order. Then the final result is generated.

4 Experimental Results

In the preprocess of the document collection, we filter out some characters, such as '-','.',':'. Then we use blank space to segment words. During indexing and searching processes, Porter stemming and stopwords removing are done. We present our experimental results as follows.

Table1 and Table2 show our experimental results for Technology Survey and Prior Art Task respectively.

Table 1. Results of TS task in terms of map and ndcg

Run	MAP	NDCG
DUT11TSRun1	0.1399	0.3876
DUT11TSRun2	0.1237	0.3704
DUT11TSRun3	0.1309	0.3824
DUT11TSRun4	0.1305	0.3319

Table 2. Resuts of PA task for 6 popular measures

Run	MAP	bpref	MRR	P_20	Recall_100	NDCG
DUT11PARun1	0.0225	0.2944	0.1417	0.0530	0.0967	0.1441
DUT11PARun2	0.0259	0.3071	0.1520	0.0630	0.1049	0.1531
DUT11PARun3	0.0323	0.3525	0.1552	0.0745	0.1198	0.1758
DUT11PARun4	0.0249	0.2993	0.1745	0.0580	0.1020	0.1503
DUT11PARun5	0.0293	0.3253	0.1827	0.0665	0.1130	0.1643
DUT11PARun6	0.0151	0.2345	0.1186	0.0310	0.0664	0.1110
DUT11PARun7	0.0305	0.3457	0.1724	0.0585	0.1307	0.1745
DUT11PARun8	0.0207	0.2953	0.1116	0.0350	0.0893	0.1410
DUT11PARun9	0.0617	0.3283	0.3529	0.1195	0.1715	0.2109

5 Conclusions

In this paper, we present our submission to the TREC 2011 Chemical IR Track. We focus on the query generation and retrieval model selection. For the Technology Survey task, we use phrase expansion method and query expansion method to generate our query, and use Query-likelihood model, DFR model and D-smoothing method to do retrieve. For the Prior Art task, we use term frequency method, tf/idf method to generate our query, and also employ the retrieval model used in TS task to execute our experiments. In our future work, we will explore more novel methods for the query patent process.

6 References

[1] Lemur Project. http://www.lemurproject.org
[2] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In Proceedings of SIGIR1998, pages: 275-281, 1998.
[3] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems, 22(2), pages: 179-214.
[4] D. Metzler, T. Strohman, H. Turtle, and W. B. Croft. Indri at TREC 2004: Terabyte Track. In Proceedings of 2004 Text Retrieval Conference(TREC 2004).
[5] G. Amati. Probabilistic models for information retrieval based on divergence from randomness. PhD thesis, Department of Computing Science, University of Glasgow, 2003.
[6] J. Gobeill, D. Teodoro, E. Patsche, P. Ruch. Report on the TREC 2009 Experiments: Chemical IR Track. In Proceedings of 2009 Text Retrieval Conference(TREC 2009)