Query Expansion with the Minimum Judgement

MASAYUKI OKABE† and SEIJI YAMADA††

Query expansion is one of feedback techniques in information retrieval, which needs a certain amount of relevance information that costs high in terms of human effort. In this paper we propose a method of query expansion which utilizes human help but with the minimum cost. Our purpose is to reduce users' cost when judging the relevancy of documents as much as possible using *Transductive Learning*. We describe this learning method is used to predict the relevancy of documents with no manual judgement based on only a fraction of true relevance information. We also show the role of the learning in our query expansion procedure. Compared with traditional query expansion methods, our method show the distinct effectiveness of query expansion, especially in the top 10 or 20 documents.

1. Introduction

Query expansion is a sort of techniques to help a user re-formulate queries in IR (Information Retrieval) systems. Many of query expansion methods use relevance information from the user to perform expansion well[1]. For those methods, the quality of query expansion significantly depends on the amount of assessed documents which are judged *relevant* or *non-relevant* by a user. However such information is generally too expensive to be elicited a lot from a user as discussed in many retrieval experiments[2]~[4]. We consider appling machine learning techniques to IR systems is one solution of the problem, especially *Transductive Learning*[5] is well-suited to the situation where a lot of relevance information cannot be expected. In contrast to normal inductive learning, the advantage of this learning method is to utilize unlabeled data for complementing the lack of labeled data. In our case, labeled data corresponds to the relevant documents given by a user, and unlabeled data corresponds to the documents whose relevancy is unknown. To utilize the similarity between labeled and unlabeled documents, this learning method try to predict the relevancy of unlabeled documents with more accuracy even based on a fraction of true relevance information.

So far, many of query expansion methods are developed, however they haven't paid much attention to the cost of relevancy judgement by a user. In this paper we present the potential performance of user feedback with the minimum judgement in the case of query expansion. Compared with traditional query expansion methods, our method differs in the procedure of finding relevant documents. Although traditional ones depend on a user efforts in this procedure, our method uses transductive learning and try to reduce much of the user's effort. Since our method extends only that procedure, it can be embedded in the most of query expansion methods.

In practical situation using relevance feedback, the initial retrieval documents usually include few relevant ones, thus effective feedback is not available in most cases. To cope with this problem, Onoda et. al[6] tried to apply one-class SVM(Support Vector Machine) to relevance feedback. Using one-class SVM, an IR system is able to utilize non-relevant documents only to improve accuracy of retrieve. This approach is similar to ours in terms of appliying a machine learning method to reduce user's cost in IR, however their study does not utilize query expansion and the minimum judgement.

The reminder of this paper is structured as follows. In section 2 we describe the two fundamental techniques for our query expansion method. In section 3 we explain the procedure of our method which builds transductive learning into normal expansion procedure. In section 4 we compared the effectiveness of our method with two other traditional query expansion methods. In section 5 we investigate the result of each topic in detail. In settion 6 we summarize our findings.

† Toyohashi University of Technology
†† National Institute of Informatics

Table 1 Term's score for query expansion

$$wpq_t = \left(\frac{r_t}{R} - \frac{n_t - r_t}{N-R}\right) * \log \frac{(r_t+0.5)/(R-r_t+0.5)}{(n_t-r_t+0.5)/(N-n_t-R+r_t+0.5)}$$

r_t	:	the number of seen relevant documents containing term t
n_t	:	the number of documents containing t
R	:	the number of seen relevant documents for query q
N	:	the number of documents in the collection

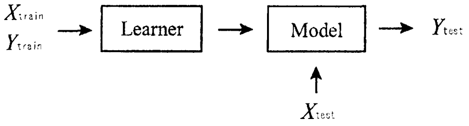

Fig. 1 Inductive Learning

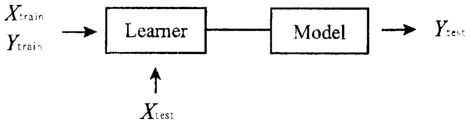

Fig. 2 Transductive Learning

2. Basic Methods

Our query expansion method is based on two basic techniques. We explain them in this section.

2.1 Query Expansion

While various query expansion techniques are proposed so far$^{7)\sim9)}$, *wpq* method is often used as a standard method$^{1),10)}$. We use this method as our basic query expansion techniques for the discussion in this paper. It calculates the score of each term appeared in relevant documents based on the formula shown in Table 1. The second term in this formula is called the Rebertson/Spark Jones weight$^{11)}$. As seen in this formula, score of a term depends on the number of relevant documents which are usually given by a user. The quality of those information affects the quality of terms selected as query expansion.

2.2 Transductive Learning

The setting of transductive learning is almost the same as the normal inductive learning. The learning task is defined on a data set X of n points $(\vec{x}_1, \vec{x}_2, ..., \vec{x}_n)$. Each data point has a disired classification label $Y = (y_1, y_2, ..., y_n)$. if we assume data points as documents, $\vec{x}_i$ is a document vector and y_i is a relevance judgement. For simplicity, we set the labels y_i are binary, i.e. $y_i \in \{+1, -1\}$. +1 and −1 means *relevant* and *non-relevant* respectively.

Normal inductive learning consists of two phases, *learning phase* and *inference phase*. In the learning phase, a subset of data points $X_{train} \subset X$ and labels $Y_{train} \subset Y$ are given as training examples. Learner will produce a model to predict labels for the rest of data points X_{test}(test examples). Using the model in inference phase, we can finally get those labels Y_{test} as shown in Fig.1.

In contrast to the inductive learning, transductive learning uses not only X_{train} but also X_{test} in the learning phase as shown in Fig.2. More over, inference phase is not separated from learning. Learning and inference are conducted at the same time in transductive setting.

Transductive learner presume labels of test examples using some similarity between labeled and unlabeled data points in order to complement the lack of training examples. Transductive learning is well-suited to the setting that $|X_{train}|$ is very small.

So far, several transductive learning methods realizing the above concept have been proposed$^{12)}$. *Spectral Graph Transducer* is one of such methods, and showed the best performance against k-nearest neighbor and transductive support vector machine. This learning method is a transductive version of k nearest-neighbor classifier, which defines the problem of labeling unlabeled examples as an optimization problem and solves it as normalised graph cuts with constraints. Since this method provides good approximation to the solution for the constrained ratiocut problem, the computational cost of learning is not so high. We use SGT as a transductive learner in our query expansion system.

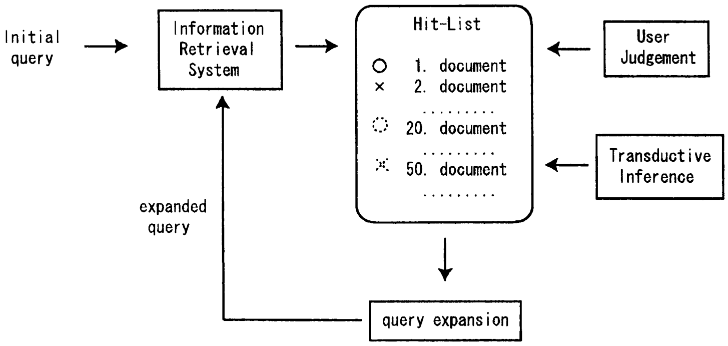

Fig. 3 Query Expansion Procedure

3. Query Expansion Procedure

Fig.3 shows the procedure of our query expansion system. We explain each step in the figure.

(1) **Initial Search**: A user gives a query to Information Retrieval System.

(2) **Judgement of Documents in Hit-List**: IR system returns a hit-list indicating the retrieval documents ranked in the ordering of relevance. A user checks the contents from the top ranked document in the hit-list, judging whether those documents are positive(relevant) or negative(non-relevant) until the user will find one each relevant and non-relevant example.

While the most of other relevance feedback and query expansion systems assumes that a user marks 'relevant' or 'non-relevant' more than 20 documents, it is very unrealistic setting because most of the users are lazy to give a mark to each document.

(3) **Learning to mark unlabeled documents**: Using transductive learning, unlabeled documents are marked their relevance judgements based on a few correct markings by the user. Our system assumes those documents which are originally unlabeled but finally labeled by transductive learning documents as correct documents when the system calculates the score of terms for expansion.

In the hit-list of Fig.3, a dashed circle of ranked in the 20th document and a dashed cross ranked in the 50th document represent judgements by transductive inference.

(4) **Select terms to expand initial query**: Based on the formula described in section 2, our system calculates the score of terms in relevant documents found in the previous step, and selects a certain number of expansion terms.

(5) **Re-input of expanded query and the next search**: New hit-list are shown to the user by inputting the expanded query.

In the above procedures, we naturally introduced transductive learning into query expansion as the effective way to automatically generate a lot of relevant documents. Thus we do not need to modify a basic query expansion procedure and can fully utilize the potentioal power of the basic query expansion.

The computational cost of transductive learning is not so much. Actually transductive learning takes a few seconds to label 100 unlabeled documents and query expansion with all the labeled documents takes also a few seconds, Thus our system can expand queries sifficiently quick in practical application.

4. Experiments

In this section, we provide empirical evidence on how our query expansion method can improve the performance of information retrieval. We compare our method with some other traditional methods.

4.1 Settings

Data set and Retrieval system We choose Okapi[13] as the retrieval system and TREC-8[14] as a data set. TREC-8 contains about 520,000 documents extracted from the database of "Foreign Register", "Financial Times", "Los Angels Times" and "Foreign Broadcast Information Service". As preprocessing, we removed stopwords and applied stemming for each document. We use 50 topics (No.401-450) from TREC-8 Ad-Hoc track as queries for evaluation. Each topic has a **TOPIC** field in its description and we use all of the terms appearing in that filed as a initial query.

We use BM25 in Okapi for the weight function in the following

$$\sum_{T \in Q} w^{(1)} \cdot \frac{(k_1+1)tf(k_3+1)qtf}{(K+tf)(k_3+qtf)} \qquad (1)$$

where Q is a query containing terms T, tf is the frequency of occurrence of the term within a specific document, qtf is the frequency of the term within the topic from which Q was derived, and $w^{(1)}$ is the Robertson/Spark Jones weight of T in Q described in section 2. In (2), K is calculated by

$$K = k_1\left((1-b) + b\frac{dl}{avdl}\right) \qquad (2)$$

where dl and $avdl$ denote the document length and the average document length measured in some suitable unit, such as word or sequence of words. In our experiments, we set $k_1 = 1.2, k_3 = 1000, b = 0.75$, and $avdl = 135.6$.

Relevance Judgements As described before, we only use the relevance information of one each top ranked relevant and non-relevant document. If we find none of relevant or non-relevant document within top 10 documents, we exclude those topics from evaluation because we cannot apply our proposed method for such topics. There are 9 exceptive topics in our experiments.

Training and Test data We use top 100 documents as *Training* and *Test* data for transductive learning. Training data is a set of two documents whose relevance information can be known manually. Remaining 98 docuemnts are the test data set whose relevancy is unknowm beforehand.

Traditional Methods We compared our query expansion method with two other methods described below.

Normal : This method simply uses only one relevant documents judged by hand. This is sometimes called *incremental relevance feedback*[2),3)].

Pseud : This method is called *pseud relevance feedback*, which assumes top n documents as relevant ones. we set 30 for n in our experiments. 30 is the best value in our preliminary experiments.

4.2 Results

Table 2 shows the precision of top $n(=10{\sim}100)$ ranked documents obtained after query expansion. Each value is averaged over topics. In this table, **sgt-0.1** represents our proposed method using spectral graph tranceducer. 0.1 is an estimate ratio of the number of true relevant documents to the whole in training and test data set. This value is one of the parameters of the sgt learning machine. We tested 0.1~0.9 for this parameter and found 0.1 is the best value in our experiment unless we change this value for each topic. Actully the optimal values are different depending on topics and the number of expansion terms. Although practically it's not easy to estimate the optimal value for each topic before query expansion, we show the result in the case that we could find the optimal value for each topic. We represent it **sgt-opt**. For comparison, we also show the performance of initial retrieval(represented as **init**) in the table.

The results show that only our method achieves significant improvements since remaining two methods show few improvements or get worse than initial retrieval. The numbers {nqe = 2,4,6,8,10} next to the name of each method in table 2 are the number of added terms to the initial query. This number does not have much effect on the performance of retrieval in each method. Our method outperformed other methods through all the cases, especially in the upper part of top 100 ranking. Since top 10 or 20 ranking is very important in practical retrieval situation, this property is useful for actual IR systems.

Table 2 Precision of Top n ranking after query expansion

		Top n Ranking									
	nqe	10	20	30	40	50	60	70	80	90	100
normal	2	0.393	0.332	0.281	0.251	0.239	0.226	0.214	0.203	0.191	0.182
	4	0.383	0.312	0.271	0.237	0.218	0.215	0.204	0.195	0.187	0.181
	6	0.393	0.304	0.267	0.241	0.216	0.198	0.188	0.180	0.172	0.166
	8	0.390	0.290	0.254	0.227	0.209	0.195	0.182	0.173	0.166	0.162
	10	0.407	0.307	0.252	0.232	0.210	0.198	0.186	0.175	0.167	0.162
pseud	2	0.405	0.348	0.308	0.273	0.250	0.236	0.218	0.205	0.198	0.191
	4	0.434	0.370	0.324	0.291	0.273	0.253	0.238	0.226	0.212	0.202
	6	0.456	0.373	0.337	0.307	0.280	0.262	0.246	0.231	0.217	0.207
	8	0.459	0.387	0.344	0.307	0.285	0.265	0.249	0.234	0.221	0.210
	10	0.461	0.376	0.330	0.304	0.287	0.264	0.252	0.239	0.228	0.217
sgt-0.1	2	0.510	0.404	0.357	0.316	0.288	0.265	0.247	0.233	0.222	0.210
	4	0.515	0.413	0.351	0.313	0.282	0.261	0.244	0.227	0.217	0.208
	6	0.532	0.423	0.352	0.314	0.292	0.270	0.248	0.234	0.219	0.209
	8	0.537	0.434	0.364	0.326	0.297	0.277	0.256	0.239	0.226	0.214
	10	0.539	0.422	0.366	0.328	0.302	0.271	0.257	0.240	0.227	0.215
sgt-opt	2	0.622	0.485	0.414	0.373	0.343	0.322	0.299	0.278	0.262	0.248
	4	0.654	**0.516**	0.440	**0.391**	**0.350**	**0.326**	**0.304**	**0.283**	**0.266**	**0.252**
	6	0.661	0.501	0.416	0.370	0.341	0.313	0.285	0.267	0.249	0.238
	8	0.661	0.507	0.429	0.387	0.346	0.315	0.287	0.267	0.252	0.240
	10	**0.673**	0.512	**0.434**	0.383	0.345	0.316	0.294	0.275	0.256	0.242
init		0.441	0.362	0.307	0.284	0.266	0.248	0.235	0.224	0.207	0.198

Table 3 The number of documents used as relevant and its accuracy

	normal	pseud	sgt-0.1	sgt-opt(2)	sgt-opt(4)	sgt-opt(6)	sgt-opt(8)	sgt-opt(10)
Used as relevant	1.00	30.00	8.93	23.81	22.10	23.54	27.49	26.95
True relevant	1.00	9.90	4.90	8.68	8.95	8.98	10.00	10.22
Accuracy	1.00	0.33	0.52	0.47	0.47	0.46	0.47	0.49

In our setting, the performance of query expanasion depends on how to assume unjudged documents as relevant ones. Obviously that the amount and the accuracy of true relevant documents are important factor to make such tentative relevant documents. Thus we investigated the number of true relevant documents and calculated its accuracy in each method. We also count the number of documents used as relevant. Table 3 shows the result(for sgt-opt(n), n is the number of expanded terms). In the table, we can see that **normal** has perfect accuracy, however its amount of true relevant documents is only one. In contrast, **pseud** has enough amount of true relevant documents, however its accuracy is low. Compared with those two methods, our method is not biased one way in the tradeoff. If we can appropriately estimate the number of true relevant documents, we can make query expansion very effective. As shown in the table, **sgt-opt** can find twice amount of true relevant documents with its accuracy keep about the same as **sgt-0.1**.

5. Discussion

We investigated the results in detail, especially how each topic changes its performance before and after query expansion. Fig.4 shows the precision improvement in the top 10 documents against initial retrieval about each topic (6 terms expanded). Horizontal line in each graph means nth topic in the TREC-8 dataset.

More than half of topics in **normal** and **psued** get worse by query expansion. Query expansion works for only a few topics. In case of **sgt**, topics can be segmented two groups. One is a group in which query expansion works very well, the other is a group in which query expansion degenerates retrieval performace. This is because misestimate of the number of true relevant document is input to the sgt learner. If we can set appropriate estimate values for the parameter, much of the second group of topics decrease as seen in the graph of **sgt-opt**.

6. Conclusion

In this paper we proposed a new query expansion method which applies transductive learn-

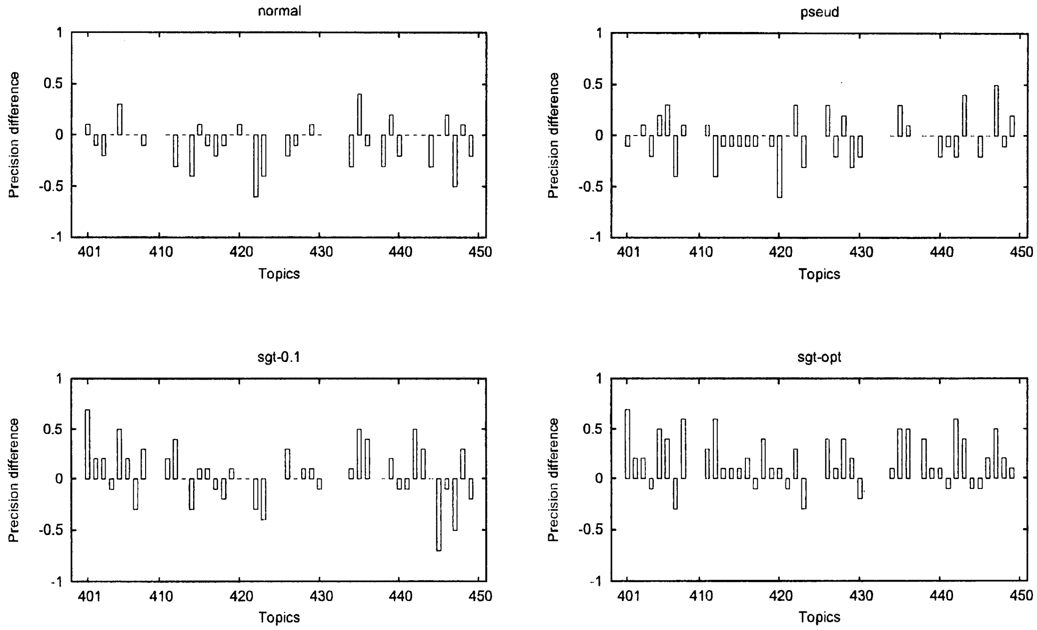

Fig. 4 Precision improvement in the the top 10 documents against initial retrieval (6 terms expanded)

ing to predict the relevancy of unlabeled documents. The experimental results showed our method outperforms initial retrieval and other traditional methods. We are now developing an algorithm to automatically estimate the number of true relevant documents in learning data sets, which is very important factor for the performance of query expansion as shown in this paper.

References

1) Ruthven, I.: Re-examining the Potential Effectiveness of Interactive Query Expansion, *SIGIR 2003 Proceedings*, ACM, pp. 213–220 (2003).
2) Aalbersberg, I. J.: Incremental relevance feedback, *SIGIR '92 Proceedings*, ACM, pp.11–22 (1992).
3) Allan, J.: Incremental relevance feedback for information filtering, *SIGIR '96 Proceedings*, ACM, pp.270–278 (1996).
4) Iwayama, M.: Relevance feedback with a small number of relevance judgements: Incremental relevance feedback vs. document clustering, *SIGIR 2000 Proceedings*, ACM, pp.10–16 (2000).
5) Vapnik, V.: Statistical learning theory, Wiley (1998).
6) Onoda, T., Murata, H. and Yamada, S.: Non-relevance feedback document retrieval, *CIS 2004 Proceedings*, IEEE, page to appear (2004).
7) Mitra, M., Singhal, M. and Buckley, C.: Improving automatic query expansion, *SIGIR '98 Proceedings*, ACM, pp.206–214 (1998).
8) Xu, J. and Croft, W. B.: Query expansion using local and global document analysis, *SIGIR '96 Proceedings*, ACM, pp.4–11 (1996).
9) Lam-Adesina, A. M. and Jones, G. J. F.: Applying summarization techniques for term selection in relevance feedback, *SIGIR 2001 Proceedings*, ACM, pp.1–9 (2001).
10) Yu, S. and et al.: Improving pseud-relevance feedback in web information retrieval using web page segmentation, *WWW 2003 Proceedings* (2003).
11) Robertson, S. E.: On term selection for query expansion, *Journal of Documentation*, Vol.46, No.4, pp.359–364 (1990).
12) Joachims, T.: Transductive learning via spectral graph partitioning, *ICML 2003 Proceedings*, pp.143–151 (2003).
13) Robertson, S. E.: Overview of the okapi projects, *Journal of the American Society for Information Science*, Vol.53, No.1, pp.3–7 (1997)
14) Voorhees, E. and Harman, D.: Overview of the eighth Text REtrieval Conference (1999).