Axiomatic Approaches to Information Retrieval - University of Delaware at TREC 2009 Million Query and Web Tracks

Wei Zheng Hui Fang
Department of Electrical and Computer Engineering
University of Delaware

Abstract

We report our experiments in TREC 2009 Million Query track and Adhoc task of Web track. Our goal is to evaluate the effectiveness of axiomatic retrieval models on the large data collection. Axiomatic approaches to information retrieval have been recently proposed and studied. The basic idea is to search for retrieval functions that can satisfy all the reasonable retrieval constraints. Previous studies showed that the derived basic axiomatic retrieval functions are less sensitive to the parameters than the other state of the art retrieval functions with comparable optimal performance. In this paper, we focus on evaluating the effectiveness of the basic axiomatic retrieval functions as well as the semantic term matching based query expansion strategy. Experiment results of the two tracks demonstrate the effectiveness of the axiomatic retrieval models.

1 Introduction

The InfoLab from the ECE department at the University of Delaware participated in both the million query track and the ad hoc task of Web track to evaluate the effectiveness of axiomatic retrieval models.

Axiomatic retrieval models have recently proposed and studied [1, 2, 3]. The basic idea is to search for retrieval functions that satisfy all of the reasonable retrieval functions. Fang and Zhai [2] derived several basic axiomatic retrieval functions and showed that they are less sensitive to the parameter setting than other existing retrieval functions with comparable optimal performance. To further improve the retrieval performance, the semantic term matching based query expansion method has also been proposed [3] in the axiomatic retrieval framework. In particular, the semantic similarity between two terms are measured with the mutual information computed over a carefully constructed working set. And the weights of semantically related terms are regulated by a set of reasonable semantic term matching constraints. It has been shown that the proposed semantic term matching method is effective to improve retrieval performance. As a query expansion method, it works equally well as the mixture language model feedback method. It is thus interesting to evaluate such a semantic term matching method in the context of the million query track.

The retrieval performance depends closely on the choice of the working set used to compute mutual information, since the working set directly affects the quality of the semantically related terms. However, it remains unclear whether the proposed semantic term matching is generally effective for all kinds of working sets and what are the root causes of the worse performance for some working sets. To better understand the above two questions, we tested the method by selecting semantically related terms from three different working sets: (1) the

working set constructed from the test collection itself, (2) the working set constructed from the Web search engine snippets; and (3) the working set constructed from the Wikipedia collection. We use the basic retrieval function as well as the expansion methods based on the three working sets in the official runs. Experiment results of MQ 09 show that the Web based method outperforms the other two and the Wikipedia based method may hinder the performance.

The paper is organized as follows. In Section 2, we explain the general idea of the axiomatic approach and semantic terms matching method. In Section 3, we describe the implementation detail of our system. The experiment results are reported and analyzed in Section 4 and the conclusions are made in Section 5.

2 Retrieval Methods

In this section, we will first explain the basic idea of the axiomatic approach and then describe the semantic term matching method.

2.1 Basic Ideas of Axiomatic Retrieval Models

Previous work [1, 2] has proposed and studied the axiomatic retrieval models, where the relevance is modeled directly with retrieval constraints. With the assumption that retrieval functions satisfying all the reasonable retrieval constraints would perform well empirically, the basic idea of axiomatic retrieval models is to search for retrieval functions that can satisfy all the retrieval constraints. Previous study derived several basic axiomatic retrieval functions [2]. Our preliminary experiments on the data collection of TREC 2008 Million Query track showed that the F2-LOG function [2] performed best, so we use this retrieval function as the baseline retrieval function in our work. The retrieval function of F2-LOG is shown as follows:

$$S(Q,D) = \sum_{t \in Q \cap D} C(t,Q) \times \frac{C(t,D)}{C(t,D) + s + \frac{s \times |D|}{avdl}} \times ln \frac{N+1}{df(t)} \tag{1}$$

where Q is the query, D is the document, $C(t,Q)$ is the term count of term t in Q, $|D|$ is the document length, $avdl$ is the average document length, N is the total number of documents and $df(t)$ is the document frequency of t.

2.2 Semantic Term Matching Method

The basic axiomatic retrieval functions rely on syntactic term matching, which can not bridge the vocabulary gaps between documents and queries. To overcome this limitation, the semantic term matching method was proposed in the axiomatic retrieval framework [3]. In particular, it allows us to incorporate the semantic similarity between query terms and terms in the document into the axiomatic retrieval functions. The method relies on three semantic term matching constraints to balance the importance of the semantic related terms and the original query terms. After incorporating the semantic term matching, the retrieval scores of a single term document $\{t\}$ for query Q can be computed based on the following functions.

$$S(Q,t) = \frac{\sum_{q \in Q} s(q,t)}{|Q|} \text{ , where } s(q,t) = \begin{cases} \omega(q) & t = q \\ \omega(q) \times \beta \times \frac{s(q,t)}{s(q,q)} & t \neq q \end{cases} \tag{2}$$

where t is a term in the document, q is a term in query Q, $\omega(q)$ is the idf of q and β is the parameter that controls how much we trust the semantically related terms. $s(q,t)$ is the semantic similarity between q and t. Note that the retrieval scores of documents with more than one

terms can be derived based on the above function as well as the document growth function discussed in the previous studies [2, 3].

The semantic similarity between terms, i.e., $s(q,t)$ is computed with the mutual information [4]:

$$s(q,t) = I(X_q, X_t|W) = \sum_{X_q, X_t \in \{0,1\}} p(X_q, X_t|W) log \frac{p(X_q, X_t|W)}{p(X_q|W)p(X_t|W)} \quad (3)$$

where X_q and X_t are two binary random variables that denote the presence/absence of query term q and term t in the document. W is the working set to compute the mutual information. We will discuss how to construct the working set in the next section.

Previous study [3] proved that the semantic retrieval function can be implemented by expanding the query with the semantic related terms and retrieve documents with basic axiomatic functions, such as F2-LOG. The weight of the expanded term can be set based on $S(Q,t)$ shown in Equation 2.

3 Implementation Details

We now describe the implementation details of our methods.

1. **Constructing working set to compute term similarity**. Equation 3 computes the semantic similarity between terms based on a working set. Previous study [3] proposed a strategy to construct a better working set based on a corpora In particular, from the corpora, the working set of a query needs to include R relevant documents and $N \times R$ random documents. In our experiments, we set R to 20 and N to 19 based on the results reported in the previous study. Note that the R relevant documents can be selected from the top R ranked documents based on the retrieval results over the corpora.

2. **Expanding query with semantically related terms**. For each query, we compute the semantic similarity between query term and terms in the working set based on Equation 3. Top K similar terms are selected for every query term. All the similar terms from a query are combined to generate the expanding term candidates for the query. The similarity between each term candidate and the whole query is computed based on Equation 2. The M most similar terms are added to the query with $S(Q,t)$ as their weights. K is set to 1000 and M is set to 20 in the experiments.

3. **Retrieving documents with expanded queries**. We can rank documents using F2-LOG retrieval function with the expanded queries.

In step 1, the semantic similarity among terms are computed based on a working set. Different working sets would give different expanded query terms. To compare the effectiveness of different working sets, we propose to use the following three working sets in the experiments.

- **Collection-based working set:** We can use the document collection itself, i.e., category B collection in MQ09, as the corpora to construct the working set. All the expanded terms will come from the document collection.

- **Wikipedia-based working set:** We can also use the Wikipedia pages, i.e., a subset of category B collection, as the corpora to construct the working set. Again, all the expanded terms will come from a subset of the document collection. Wikipedia pages are manually written and compiled, they contain knowledge contributed by different people. Intuitively, the quality of a Wikipedia page should be higher than a random web

page. Compared with the collection-based working set, this working set might be able to contribute more high quality semantically related terms if there exist Wikipedia pages that are relevant to a query.

- **Web-based working set:** Another possible solution is to construct the working set based on the data from Web. In particular, we submit queries to a leading Web search engine and construct the working sets based on the top 100 returned snippets. Clearly, the data indexed by the web search engine should be much larger then the category B collection. Thus, we expect this working set would contribute more semantically related terms that can not be discovered in the document collection.

4 Experiment results

4.1 Results of Official Runs

We submitted five runs to the million query track and three runs to the ad hoc task of Web track. Here are the descriptions of the submitted runs.

1. **UDMQAxBL/UDWAxBL:** These are our baseline methods. F2-LOG function is used as the retrieval function. UDMQAxBL is the run submitted to MQ track and UDWAxBL is the run submitted to Adhoc task of Web track.
2. **UDMQAxBLlink:** We use both the anchor text and document content to rank documents. F2-LOG function is used as the retrieval function.
3. **UDMQAxQE/UDWAxQE:** This run uses the semantic term matching method, and the semantically related terms are selected from the collection-based working set.
4. **UDMQAxQEWP:** This run uses the semantic term matching method, and the semantically related terms are selected from the Wikipedia-based working set.
5. **UDMQAxWeb/UDWAxWeb:** This run uses the semantic term matching method, and the semantically related terms are selected from the Web-based working set.

The official runs are evaluated with both statMAP and MTC measures [5]. In the tracks, the default value of β in Equation 2 of semantic matching method is 1.5 according to the training result on Robust04 collection.

Table 1 summaried the results of our five official runs in MQ track. The *eMAP.base* and *statMAP.base* in the table denoted the eMAP and statMAP value for queries that our runs contributed to. The *eMAP.reuse* and *statMAP.reuse* denoted the eMAP and statMAP for queries that our runs were held out from. We can see that the semantic term matching method can significantly improve the performance. In all evaluations of Table 1, the semantic matching with web collection performed best. The semantic matching with Wikipedia often performed worse than the baseline. The reason is that it can introduce many noise into the original query if there is no related Wikipedia page for the query.

Table 2 listed the eMAP values of MTC evaluation and statMAP values in official reported results of Web track. The basic axiomatic function F2-LOG had the highest eMAP value and query expansion with web collection had the highest statMAP value in our three runs. Note that the queries used in Web track is a subset of those used in MQ track, it is interesting to see that the performance comparison among different methods are not consistent with what we observed in MQ results.

Table 1: Results of official runs for Million Query track

	AxBL	**AxBLlink**	**AxQE**	**AxQEWp**	**AxQEWeb**
eMAP.base	0.08111	0.05890	0.09637	0.082188	**0.1148**
eMAP.reuse	0.07142	0.05449	0.07316	0.061861	**0.09238**
statMAP.base	0.25498	0.19178	0.20536	0.13347	**0.27147**
statMAP.reuse	0.22081	0.15353	0.13453	0.087670	**0.22976**

Table 2: Results of our official runs in Adhoc task of Web track

	AxBL	**AxQE**	**AxQEWeb**
eMAP	**0.04245**	0.02401	0.03838
statMAP	0.17583	0.13329	**0.19986**

4.2 Results for different query categories

We now evaluate the performance of all the five runs in MQ based on query categories. Table 3 reportd the performance based on different query categories. Although the results were similar to what we observed from Table 1, we can make two interesting observations. First, although AXQEWp hurted the performance in most bases, it can improve the retrieval performance over the baseline for hard queries. Second, AXQEWeb can significantly improve the retrieval performance in most cases, but it can not improve the performance for easy queries. Clearly, it suggests that if we could predict query categories correctly, we might be able to get better performance by using different retrieval strategies for different query categories.

Table 4 showed the top 10 expanded terms of the semantic term matching methods in different query categories. In the query 20215, the AP values of the original query, AxQE, AxQEWp and AxQEWeb are 0.296, 0.206, 0.01, 0.313. We can see that the AxQEWeb method found a lot of semantic related terms, such as *abuse, false, lawsuit, overbil*, etc., to expand the query. The AxQE and AxQEWP selected some semantic related terms, such as *abus, inspector* and *claim*, but they also included a lot of noisy terms, such as *beneficiary*, *byrd*, and *dingell*. These noisy terms made the expanded query to perform worse than the original query.

5 Conclusions

We evaluate the axiomatic retrieval models in the context of the million query track and the ad hoc task of Web track. Both the basic axiomatic retrieval function and the semantic term matching strategy have been shown to be effective based on the results. Moreover, we compare the effectiveness of three working sets which are used to compute the semantic similarities among terms. Different working sets would lead to different expanded terms. Experiment

Table 3: Performance comparison for different query categories (measured with statMAP)

	AxBL	**AxBLlink**	**AxQE**	**AxQEWp**	**AxQEWeb**
EASY	**0.44879**	0.32769	0.35647	0.19644	0.43225
MEDIUM	0.17732	0.15060	0.12315	0.11231	**0.21652**
HARD	0.03835	0.04882	0.02786	0.05242	**0.10254**
	AxBL	**AxBLlink**	**AxQE**	**AxQEWp**	**AxQEWeb**
PREC	0.22435	0.16076	0.15830	0.11162	**0.25343**
RECALL	0.23868	0.19817	0.18954	0.13323	**0.26402**

Table 4: Expanded queries in different query categories

	Origin	**AxQE**	**AxQEWp**	**AxQEWeb**
EASY	query 20643: architects in new jersey	aia, architecure, associate, building, construct, englewood, firm, flemington, hoboken, interior	american, building, connecticut, construct, design, dome, firm, georgia, illinoi, indiana	ahm, architecture, construct, design, institute, license, lopatcong, millburn, nj, stryker
MEDIUM	query 20215: report medicare fraud	abuse, agency, beneficiary, bill, care, department, fraudulent, health, inspector, medicaid	amendment, byrd, care, claim, deficit, dingell, federal, health, healthcare, intermediary	abuse, attempt, cohen, false, fraudulent, lawsuit, medicaid, medicare, million, overbilled
HARD	query 20101: join job corps	chart, deployment, duty, enlist, ethics, humor, igoogl, insignia, legislative, marine	care, career, civilian, cloth, employ, force, home, ii, labor, occupation	education, invite, largest, nation, recruit, resident, train, visit, vocation, web

results show that the Web-based working set is the most effective one, while the Wikipedia-based working set is the least effective one. Our analysis suggests that the worse performance of Wikipedia-based working set is probably caused by the noises introduced by the Wikipedia when there is no wiki pages related to the given query.

References

[1] H. Fang, T. Tao, and C. Zhai. A formal study of information retrieval heuristics. In *Proceedings of SIGIR-04*, 2004.

[2] H. Fang and C. Zhai. An exploration of axiomatic approaches to information retrieval. In *Proceedings of SIGIR-05*, 2005.

[3] H. Fang and C. Zhai. Semantic Term Matching in Axiomatic Approaches to Information Retrieval. In *Proceedings of SIGIR-06*, 2006.

[4] C. J. Van Rijsbergen. Information Retrieval, 1979.

[5] B. Carterette, V. Pavlu, H. Fang and E. Kanoulas. Million Query Track 2009 Overview. In *Proceedings of TREC-09*, 2009.