

Cross-Corpus Relevance Projection

Nima Asadi
Dept. of Computer Science
University of Maryland
College Park, MD
nima@cs.umd.edu

Donald Metzler
Information Sciences Institute
Univ. of Southern California
Marina del Rey, CA
metzler@isi.edu

Jimmy Lin
The iSchool
University of Maryland
College Park, MD
jimmylin@umd.edu

Categories and Subject Descriptors: H.3.3 Information Storage and Retrieval: Information Search and Retrieval

General Terms: Algorithms, Experimentation

Keywords: test collections, learning to rank, web search

1. INTRODUCTION

Document corpora are key components of information retrieval test collections. However, for certain tasks, such as evaluating the effectiveness of a new retrieval technique or estimating the parameters of a learning to rank model, a corpus alone is not enough. For these tasks, queries and relevance judgments associated with the corpus are also necessary. However, researchers often find themselves in scenarios where they only have access to a corpus, in which case evaluation and learning to rank become challenging.

Document corpora are relatively straightforward to gather. On the other hand, obtaining queries and relevance judgments for a given corpus is costly. In production environments, it may be possible to obtain low-cost relevance information using query and click logs. However, in more constrained research environments these options are not available, and relevance judgments are usually provided by humans. To reduce the cost of this potentially expensive process, researchers have developed low-cost evaluation strategies, including minimal test collections [2] and crowdsourcing [1]. Despite the usefulness of these strategies, the resulting relevance judgments cannot easily be “ported” to a new or different corpus.

To overcome these issues, we propose a new method to reduce manual annotation costs by transferring relevance judgments across corpora. Assuming that a set of queries and relevance judgments have been manually constructed for a *source* document corpus \mathcal{D}_s , our goal is to automatically construct a test collection for a *target* document corpus \mathcal{D}_t by *projecting* the existing test collection from \mathcal{D}_s onto \mathcal{D}_t .

The goal of projecting test collections is not to produce manual quality test collections. In fact, it is assumed that projected test collections will contain noisy relevance judgments (i.e., ones which humans are unlikely to agree with). The important question, however, is whether these noisy projected judgments are *useful* for training ranking models in the target corpus.

2. PROJECTED TEST COLLECTIONS

As mentioned earlier, the input to our proposed model consists of a source document corpus \mathcal{D}_s along with a set of existing queries and relevance judgments for that corpus. It is assumed that these judgments are provided in the form of “relevance” (Q, D, R) tuples, i.e., (query, document, relevance). Our goal is to leverage the existing judgments for corpus \mathcal{D}_s and create tuples of the same form for the target corpus \mathcal{D}_t for the same set of queries. In this way, we use the same set of queries across test collections, but find new documents and estimate their relevance in the target corpus.

We group existing tuples based on their Q field and perform the procedure described below for each of the groups separately. In other words, we project the relevance judgments for each query separately.

Using a set of features extracted from existing relevance tuples for a query, we train what we refer to as a supervised “relevance classifier”. This classifier is capable of predicting relevance scores for a given document with respect to the query for which the classifier is designed. It is important to note that the type of features used to train the classifiers as well as the learning model have a significant impact on the performance of the relevance classifiers. Therefore, it is important to choose an expressive set of features, such as those commonly used in learning to rank models.

Once a classifier is trained over the source corpus for a query, the next step is to find a set of documents from the target corpus that can be labeled by the relevance classifier. Unlike the pooling method, this set of documents does not necessarily have to be the output of a given retrieval system. However, for convenience, we use BM25 to retrieve 1000 candidate documents per query for automatic labeling.

The final task is to assign relevance labels to the set of documents found in the target corpus \mathcal{D}_t . Even though research has shown the usefulness of graded relevance judgments, it makes the most sense for our relevance classifiers to assign binary judgments (i.e., “relevant” and “non-relevant”). However, it might be desirable for each label to have an associated confidence score. These scores may be informative for evaluation or learning to rank.

After repeating the above steps for all queries from a source corpus, a complete set of relevance tuples is constructed for the target corpus. It is important to note that since projections are carried out per query, there is no restriction on the number of source corpora that can be used to construct test collections for a target corpus. That is to say, a set of relevance judgments can be constructed for a query, given the relevance classifier designed for that query,

		Target					
		Wt10g		Gov2		ClueWeb09	
		<i>proj</i>	<i>base</i>	<i>proj</i>	<i>base</i>	<i>proj</i>	<i>base</i>
Source	Wt10g	-	-	.181	.179	.097	.098
	Gov2	.127	.125	-	-	.087	.088
	ClueWeb09	.126*	.120	.168	.179	-	-

Table 1: Effectiveness (ERR@20) of training on the projected (*proj*) test collections versus the baseline models (*base*).

regardless of the source corpus. This provides a mechanism for accumulating projected judgments from different sources for a fixed target test collection.

3. EXPERIMENTAL EVALUATION

We performed experiments using three TREC datasets: Wt10g (1.6m pages, topics 451–550), Gov2 (25m pages, topics 701–850), and the first English segment of ClueWeb09 (50m pages, topics 1–100). Title queries are used in all cases.

Our relevance classifier uses a maximum entropy model and is trained on a feature set consisting of basic information retrieval scores (language modeling and BM25 scores), and term proximity features (exact phrases, ordered windows, unordered windows, etc.).

The criterion we use to determine if our proposed approach is successful or not is whether a learning to rank model trained using a projected test collection over the target corpus can achieve better performance than a learning to rank model trained on the source corpus and tested on the target corpus. If this is the case, then we have shown that our projection method can be used to effectively transfer judgments across corpora. The effectiveness of the learned models is measured using manual judgments from each TREC test collection described above, using Expected Reciprocal Rank (ERR). To determine if differences between the various models are statistically significant, we utilize a one-side paired *t*-test.

We use a relatively straightforward learning to rank model in our experiments: an iterative greedy feature selection strategy that directly optimizes ERR to produce a linear ranking function, similar to the one described by Metzler [3]. Note that our ranking model shares the same features as the relevance classifier. This is a deliberate choice, but the alternative approach (disjoint features) may also have merits.

Table 1 shows the effectiveness (ERR@20) of the models trained using the projected test collections. It also illustrates the effectiveness of the baseline systems across different corpora. As described previously, the baseline is a ranking model trained using manual judgments from the source corpus and evaluated using the target corpus—in other words, no domain adaptation (but otherwise using exactly the same features and same learning method). The differences are not statistically significant in most cases, which means that relevance projection provides little benefit. However, the projected conditions fared no worse either, which suggests that despite the noisy projection process, relevance signals from the source corpus are largely preserved in the target corpus.

However, an opportunity that our proposed method creates is the ability to project judgments from *multiple* sources onto a *single* target corpus. Table 2 shows the effectiveness of projecting from two different sources onto each target corpus. Here, the baseline is simply joining the feature vectors

Sources→Target	Baseline	Projection
(1) Gov2,Wt10g→ClueWeb09	0.091	0.101*
(2) ClueWeb09,Wt10g→Gov2	0.174	0.182*
(3) ClueWeb09,Gov2→Wt10g	0.122	0.127

Table 2: Effectiveness (ERR@20) of training on projections from multiple sources.

from the two source collections to train the model, and then testing on the target corpus.

Results show that the somewhat naïve baseline of taking the union of training data from the two sources works rather poorly—worse than simply training on the “better” of the two sources alone. However, it is important to note that in a real-world scenario, we wouldn’t have access to test data in the target corpus, so there would be no way of knowing which source was “better”.

We see that models learned by projecting from Gov2 and Wt10g onto ClueWeb09 (1) and from ClueWeb09 and Wt10g onto Gov2 (2) are significantly better than the baseline (as denoted by asterisks in Table 2). In the case of (1), the projection condition is significantly better than training on Gov2 alone and testing on ClueWeb09, but not true for Wt10g. In the case of (3), the projection condition is significantly better than training on ClueWeb09 alone and testing on Wt10g, but not true for Gov2.

Although the results are somewhat inconclusive, we find that our proposed approach for projecting relevance judgments yields reasonable quality training data in the target corpus. Our method also allows more than one source to be projected onto the same target, and hints that more, but possibly lower-quality, training data in some cases leads to significant improvements.

4. CONCLUSIONS AND FUTURE WORK

We proposed a classification-based approach to transfer existing relevance judgments from one or more source corpora to a target corpus. The resulting projected test collection, which contains the same queries as the source, can be used for evaluation and to train learning to rank models. Experimental evaluations show that when projecting multiple source corpora onto a target corpus, it is sometimes possible to obtain significant improvements. Although the results are somewhat inconclusive, this finding is nevertheless interesting, and encourages further study in a few directions: different classifiers, different feature sets, and a different set of candidate documents in the target corpus on which the classifier is applied.

5. REFERENCES

- [1] O. Alonso, D. Rose, and B. Stewart. Crowdsourcing for relevance evaluation. *SIGIR Forum*, 42(2):9–15, 2008.
- [2] B. Carterette. *Low-Cost and Robust Evaluation of Information Retrieval Systems*. PhD thesis, University of Massachusetts, 2008.
- [3] D. Metzler. Automatic feature selection in the Markov random field model for information retrieval. In *CIKM 2007*, pages 253–262.