# The University of Massachusetts Amherst's participation in the INEX 2011 Prove It Track

Henry A. Feild, Marc-Allen Cartright, and James Allan

Center for Intelligent Information Retrieval
University of Massachusetts Amherst, Amherst MA 01003, USA
{hfeild, irmarc, allan}@cs.umass.edu

**Abstract.** We describe the process that led to the our participation in the INEX 2011 Prove It task. We submitted the results of six book page retrieval systems over a collection of 50,000 books. Two of our runs use the sequential dependency model (a model that uses both unigrams and bigrams from a query) and the other four interpolate between language model scores at the passage level and sequential dependency model scores at the page level. In this report, we describe our observations of these and several other retrieval models applied to the Prove It task.

**Keywords:** INEX, Prove It, Book Retrieval, Sequential Dependency Model, Passage Retrieval

## 1 Introduction

In this report we describe our submissions to the 2011 INEX Prove It task, where the goal is to rank book pages that are supportive, refutative, or relevant with respect to a given fact. We did not participate in the optional sub task of classifying each result as confirming or refuting the topic; in our submissions we labeled all retrieved documents as *confirming* the fact. To determine what retrieval systems to submit, we investigated several models. In the following sections, we detail those models and give a summary of the results that led to our submissions.

## 2 Indexing and Retrieval Models

We only considered indexing pages. The index used no other information about a page's corresponding book, chapter, or section and all tokens were stemmed using the Porter stemmer. We indexed a total of 6,164,793,369 token occurrences from 16,971,566 pages from 50,232 books using a modified version of the Galago retrieval system.[1]

We explored a number of models for page and passage retrieval, including relevance modeling, sequential dependence modeling, passage modeling, stop word removal, and mixtures thereof. We describe each below.

[1] http://galagosearch.org/

**Query likelihood language modeling (QL).** This model scores each page by its likelihood of generating the query [4]. The model also smooths with a background model of the collection; for this, we used Dirichlet smoothing with the default smoothing parameter: $\mu = 1500$.

**Relevance modeling (RM).** A form of pseudo relevance feedback, relevance modeling creates a language model from the top $k$ pages retrieved for a query, expands the query with some number of the most likely terms from the model, and performs a second retrieval [2]. We investigated relevance modeling because, as with all pseudo relevance feedback methods, it allows the vocabulary of the original query to be expanded, hopefully capturing related terms. There are three parameters to set: the number of feedback pages to use (set to 10), the number of feedback term to use (also set to 10), and the weight to give the original query model and the relevance model for the second retrieval (set to 0.5). These are the default settings distributed with Galago.

**Sequential dependence modeling (SDM).** This model interpolates between document scores for three language models: unigram, bigram, and proximity of adjacent query term pairs [3]. Because of its use of bigrams, SDM captures portions of phrases that unigram models miss. The weight of each sub language model are parameters, and we used the defaults suggested by Metzler and Croft [3]: 0.85, 0.10, 0.05 for the unigram, bigram, and proximity models, respectively. In addition, we used Dirichlet smoothing for each language model and experimented with $\mu = 1500$ (the Galago default) and $\mu = 363$ (the average number of terms per page).

**Passage modeling (PM).** This model first scores passages using QL with Dirichlet smoothing (setting $\mu$ to the length of the passage), selects the highest passage score per page, and then interpolates between that score and the corresponding page's SDM score. In our implementation, the top 1,000 pages (*Pages*) and the top 10,000 passages (*Pass*)[2] are retrieved as two separate lists and then interpolated. If a passage is present in *Pass*, but the corresponding page is not in *Pages*, the page score is set to the minimum page score in *Pages*. Likewise, if a page is retrieved in *Pages* but no passages from that page are present in *Pass*, the lowest passage score in *Pass* is used as a proxy. The parameters of the PM model include the passage length $l$ and the interpolation factor, $\lambda$, where the maximum passage score is weighted by $\lambda$ and the page score is weighted by $1 - \lambda$. We experimented with several values of $\lambda$.

**Stop word removal (Stop).** When stopping is used, query terms found in a list of 119 stop words[3] are removed.

We considered several combinations of the above models, all using stemming. These include: LM, RM, SDM, SDM+RM, PM, and each of these with and without stop words removed.

[2] We allow multiple passages per document to appear on this list; filtering the highest scoring passage per page is performed on this 10,000 passage subset.

[3] http://www.textfixer.com/resources/common-english-words.txt

| Field | Example 1 | Example 2 | Example 3 |
|---|---|---|---|
| ID | 2010000 | 2010012 | 2010015 |
| Fact | In the battle of New Orleans on the 8th of January 1815, 2000 British troops were killed, wounded or imprisoned, while only 13 American troops were lost, 7 killed and 6 wounded. | The main function of telescope is to make distant objects look near. | Victor Emanuel enters Rome as king of united Italy. |
| Info need | All sections of books that detail the losses suffered either at the British or the American side are relevant. I am not interested in how the battle was fought, but just want to find out about the losses at the end of the battle. | Most of the book is relevant to Astronomy as its a handbook on astronomy. | Italy will celebrate next year its re-unification and I needed to check the facts and their dates. Italy had two other capitals, Turing and Florence, before it was possible to get Rome back from the Vatican State. |
| Query | New Orleans battle 1815 troops lost killed | Telescope | Rome capital |
| Subject | battle of New Orleans 1815 | telescope | Rome becomes capital of united Italy |
| Task | My task is to find out the scale of losses on both the British and American side in the battle of New Orleans in 1815 | We need to write a primer on Astronomy. | Find out the date when Rome became capital of reunited Italy. |

**Table 1.** The INEX Prove It topic fields and examples.

## 3 Training data

Of the 83 total topics available for the Prove It task, 21 have judgments to evaluate submissions from the 2010 INEX Prove It workshop. We used these as the basis for our training set.

Inevitably, new systems pull up unjudged book pages in the top ten ranks. To handle these cases, we developed a judgment system with which lab members, including the authors, annotated pages as being *supportive*, *refutative*, or *relevant* in the case that a page was on topic, but not distinctly and completely supportive or refutative. The system displayed all fields of a topic, making the annotator as informed as possible. The fields are listed in the first column of Table 1 along with three examples of the field contents. The *info need* field usually describes what should be considered relevant, and the accessors were asked to abide by this. Some topics were tricky to judge, as in the case of Example 3 in Table 1 (Topic 2010015), where the broad focus is clearly on Italy, but the specific information being sought is inconsistent across the fields. In cases such as these, annotators were asked to interpret the information need as best they could and judge all pages relative to that interpretation.

Using the procedure described above, we augmented our training set with 535 additional relevance judgments. This covers many of the unjudged documents the systems retrieved in their top 10 lists for each topic.

|  | NDCG@10 |  |
|---|---|---|
| System | Stopped | Unstopped |
| $\mathrm{LM}_{\mu=1500}$ | 0.811 | 0.811 |
| RM | 0.751 | 0.701 |
| $\mathrm{SDM+RM}_{\mu=1500}$ | 0.755 | 0.751 |
| $\mathrm{SDM}_{\mu=1500}$ | 0.834 | 0.854 |
| $\mathrm{SDM}_{\mu=363}$ | 0.828 | 0.854 |
| $\mathrm{PM}_{l=100,\lambda=0.25}$ | 0.856 | 0.859 |
| $\mathrm{PM}_{l=50,\lambda=0.25}$ | 0.863 | 0.873 |

**Table 2.** The results of several systems over the 21 training topics.

## 4 Results

In this section we discuss the performance of the models listed in Section 2 on the training data and our submitted models on the INEX 2011 test data.

### 4.1 Results over training topics

We evaluated over all 21 training topics. Each model considered only the *fact* field of each topic; when using the *query* field, our best models only outperform the better systems from last year's track by a small margin [1]. The substantial difference in performance between using the the two fields appears to stem from the poor representation of the information need in most topics' *query* field. Consider Example 2 in Table 1: the query "telescope" does not adequately describe the information need, which is the assertion that the primary function of a telescope is to magnify distant objects.

Table 2 reports the normalized discounted cumulative gain at rank 10 (NDCG-@10) of the systems with and without stopwords removed. We binarized the graded relevance judgments such that the *supportive*, *refutative*, and *relevant* labels are conflated. The relevance models do not perform as well as the others, though this is partially due to not having enough judgments. Even if the unjudged documents are assumed relevant, SDM outperforms RM in the unstopped case, and RM only marginally improves over SDM in the stopped case. Setting $\mu$ to the average page length was not helpful for SDM, however, we entered $\mathrm{SDM}_{\mu=363}$ as a submission because without a comprehensive parameter sweep, setting $\mu$ to the average page length is more principled than the Galago default.

The PM models outperform the others, with a passage size of 50 terms taking the lead. To understand why we choose $\lambda = 0.25$,[4] see Figure 1 (this only shows the variation with stop words removed). For both 50 and 100 term passages, it is clear that a value of $\lambda$ in the $[0.20, 0.30]$ range, and specifically 0.25, is optimal. This places much of the final page score on SDM, but still gives a substantial amount of weight to the maximum LM passage score.

SDM captures pieces of phrases in a fact, and these seem to be important given the results. PM adds the notion of tight proximity—a high passage score

[4] Our submissions' names suggest we used $\lambda = 0.025$, however this was a typo.

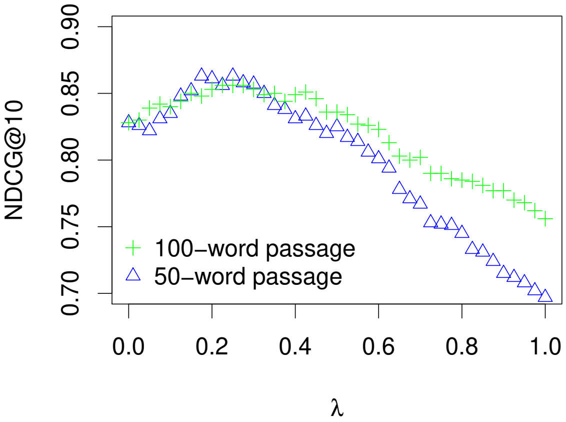

**Fig. 1.** A sweep over the $\lambda$ parameter for the passage model. Smaller $\lambda$ values mean more weight is given to the SDM score of the page, while higher values mean more weight is given to the highest scoring passage (using QLM). All queries were stopped.

ideally applies to passages that are topical hot spots. By setting $\lambda = 0.25$, the model ranks pages that seem relevant overall and also contain topical hot spots higher than those that do not, which means the page's content is more important than the content of any single passage. Said differently, a page with many medium scoring passages will be ranked higher than a page with one high scoring passage. We performed a manual inspection of retrieved documents and found pages that have only one high scoring passage are often non-relevant. The passage may make reference to an aspect of the topic, but provides no in depth information. Perhaps due to the nature of books, relevant sections tend to discuss topics over several paragraphs and even pages. Thus, the behavior of PM when $\lambda = 0.25$ is consistent with our observations of relevance within the data set.

### 4.2 Results over test topics

INEX participants provided a limited number of judgments for nine topics. These judgments cover only about 20% of the the top ten pages retrieved across the 18 submitted runs. The mean average precision (MAP), mean reciprocal rank (MRR), precision at 10 (P@10) and NDCG@10 are reported for each of our submissions in Table 3. The limited number of judgments is apparent in the lower NDCG@10 figures. The results suggest that removing stop words is detrimental, which is consistent with our findings with the training data. The two best performing runs are SDM with $\mu = 363$ and PM with 100 word passages and $\lambda = 0.25$, however, the 50-word passage model was not far behind. Overall, our models performed very well, but more judgments are necessary to fully understand the differences among them.

| System | Stopped | MAP | MRR | P@10 | NDCG@10 |
|---|---|---|---|---|---|
| $\text{SDM}_{\mu=363}$ | no | **0.2039** | 0.3890 | **0.1556** | **0.2768** |
| $\text{SDM}_{\mu=363}$ | yes | 0.1752 | 0.3220 | **0.1556** | 0.2437 |
| $\text{PM}_{l=50,\lambda=0.25}$ | no | 0.2037 | 0.3889 | **0.1556** | **0.2768** |
| $\text{PM}_{l=50,\lambda=0.25}$ | yes | 0.1743 | 0.3223 | **0.1556** | 0.2437 |
| $\text{PM}_{l=100,\lambda=0.25}$ | no | 0.2035 | **0.3894** | **0.1556** | **0.2768** |
| $\text{PM}_{l=100,\lambda=0.25}$ | yes | 0.1740 | 0.3236 | **0.1556** | 0.2447 |

**Table 3.** The results of our submissions on the nine INEX 2011 test queries. Best results are shown in bold.

## 5 Summary

We considered several systems to retrieve supportive and refutative book pages for a given fact as part of the 2011 INEX Prove It task. We found that sequential dependence modeling (SDM) and passage-page interpolation (PM) perform best. Based on the behavior of these two systems and our observations of relevance from a manual inspection, relevant book pages tend to discuss the relevant material across many paragraphs. While PM attempts to model this to some degree, we believe that this phenomenon can be modeled in more powerful ways, which we leave to future work.

## 6 Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, in part by NSF CLUE IIS-0844226 and in part by NSF grant #IIS-0910884. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect those of the sponsor.

## References

1. Geva, S., Kamps, J., Schenkel, R., Trotman, A. (eds.): Pre-proceedings of the INEX workshop (2010)
2. Lavrenko, V., Croft, W.: Relevance based language models. In: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 120–127. ACM (2001)
3. Metzler, D., Croft, W.B.: A markov random field model for term dependencies. In: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. pp. 472–479. SIGIR '05, ACM, New York, NY, USA (2005)
4. Ponte, J., Croft, W.: A language modeling approach to information retrieval. In: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval. pp. 275–281. ACM (1998)