Improving Search Effectiveness with Field-based Relevance Modeling

Binsheng Liu
RMIT University
Melbourne, Australia

Xiaolu Lu
RMIT University
Melbourne, Australia

Oren Kurland
Technion – Israel Institute of Technology
Haifa, Israel

J. Shane Culpepper
RMIT University
Melbourne, Australia

ABSTRACT

Fields are a valuable auxiliary source of information in semi-structured HTML web documents. So, it is no surprise that ranking models have been designed to leverage this information to improve search effectiveness. We present the first (initial) study of utilizing field-based information in the relevance modeling framework. Fields play two different, and integrated, roles in our models: sources of information for inducing relevance models and units on which relevance models are applied for ranking. Our preliminary results suggest that field-based relevance modeling can improve precision at top ranks; specifically, to a greater extent than the commonly used BM25F and SDM-Fields field-based models. Further analysis shows that using field-based relevance models mainly improves the effectiveness of tail queries. Our findings suggest that using field-based information together with relevance modeling is a promising area of future exploration.

KEYWORDS

relevance modeling; web search; field-based retrieval models

1 INTRODUCTION

The match between a query and document fields — e.g. `title`, `heading`, and `inlink` which are derived from HTML and SGML markups — is often assumed to be a strong relevance signal. Indeed, commonly used Web retrieval methods such as BM25F [13] and SDMF [7] outperform their non-field counterparts.

Existing field-based retrieval methods are mainly based on surface-level field-query comparisons [7, 13]. However, short fields (e.g., titles) of relevant documents are prone to increased vocabulary mismatch with the query. One of the most fundamental and principled paradigms to addressing the vocabulary mismatch problem for whole documents is relevance modeling [5]: terms in the query and in relevant documents are assumed to be generated by a latent relevance language model. A relevance model is usually induced from pseudo relevant documents — i.e., those most highly ranked by initial search. The comparison between the relevance model and document language models serves for ranking.

Past work on relevance modeling has focused on unstructured text. We present a study of using field-based information in the relevance-modeling framework. Our first method induces relevance models from fields independently and then linearly combines them to create a weighted model. The second method is based on inducing a relevance model from the entire document and using it to score fields. Hence, fields are used in two (integrated) capacities: sources of information for inducing relevance models and units scored by using relevance models.

Another important aspect of our work is a comprehensive failure analysis of field-based retrieval performance when applying relevance modeling. While past work has demonstrated the *average* effectiveness of field-based methods, failure analyses are rare. These are important for shedding light on potential avenues for performance improvement.

Experiments performed using TREC's ClueWeb09 collection show that while common field-based ranking models improve early precision effectiveness, using field-based information with relevance models can further improve it; specifically, with respect to using relevance modeling with whole documents as is the standard. Our failure analysis shows that field-based relevance modeling is mainly effective for tail queries. We also show that using field-based information, with or without relevance modeling, has mixed effects in terms of mean average precision (MAP).

2 RELATED WORK

Our focus is on integrating field-based information and relevance modeling. Thus, we briefly review commonly used field-based retrieval methods and relevance modeling for whole documents [5].

BM25F. *BM25F* extends *BM25* [10] by incorporating field information directly into the ranking function. Robertson et al. [11] proposed to boost the weights of terms that also appeared in fields. Zaragoza et al. [13] combined the normalized weighted term frequency to produce the BM25F document scoring function.

SDMF. *SDM Fields* is another state-of-the-art retrieval technique for Web data which has been shown to work well on the ClueWeb09B collection [3, 6]. The model extends SDM [6] to fields. The unigram, unordered bigram, and ordered bigram are scored on fields respectively and combined. In practice, the unordered bigram and ordered bigram models are only applied on the `body` field.

ADCS '18, December 11–12, 2018, Dunedin, New Zealand

ACM ISBN 978-1-4503-6549-9/18/12…$15.00
https://doi.org/10.1145/3291992.3292005

Field-based Language Models. Ogilvie and Callan [9] used fields as document representations in the query likelihood model:

$$P(q|d) = \prod_{t \in q} \sum_{f} W_f P(t|f,d); \qquad (1)$$

$P(q|d)$ is the probability of generating q's terms by a language model induced from document d; $P(t|f,d)$ is the probability assigned to term t by a language model induced from field f in d; W_f is f's weight.

Kim and Croft [4], as us, used relevance models induced from fields. However, a significant fundamental difference with our work is that the relevance models were not used to directly score fields or the entire document as implied by the generative theory for relevance. Rather, the relevance models were used to assign a weight $W_{f,t}$ for each field f with respect to each query term t ($\in q$). These weights were then used in the query-likelihood retrieval model from Equation 1 instead of W_f which is the same weight for all query terms with respect to f. Specifically, $W_{f,t}$ is the normalized probability assigned to term t by a relevance language model induced from field f of top-retrieved documents; normalization is with respect to all fields. Thus, the suggested retrieval model is still based on scoring a query w.r.t. a field using the surface-level similarity between the two. In contrast, our models try to alleviate the vocabulary mismatch problem incurred by such scoring by using relevance models to score the fields. We note that using relevance-model-based field weights in our models is an interesting avenue for future work.

Zamani et al. [12] explored incorporating document fields into neural ranking models. They use n-grams to represent field information, build neural models for each field and then ensemble all of the models to obtain the final ranking.

As already noted, relevance models are usually induced using pseudo feedback. Specifically, let D_{QL} be the list of the k documents most highly ranked by the query likelihood model. Then, relevance model #1 (RM1) is estimated as:

$$P(t|RM1) = \sum_{d \in D_{QL}} P(t|d) P(d|q); \qquad (2)$$

t is a term; $P(t|d)$ is the probability assigned to t by a language model induced from document d; $P(d|q)$ is d's normalized (over D_{QL}) query likelihood. To alleviate query drift, RM1 is anchored to the original query using a free parameter λ, yielding RM3 [1]:

$$P(t|RM3) = \lambda P(t|q) + (1-\lambda) P(t|RM1). \qquad (3)$$

3 OUR APPROACH

In this section, we present our methods that integrate field-based information and relevance modeling. The first method induces relevance models from fields and scores the fields independently. The second method scores fields using a relevance model induced from the entire document.

3.1 Relevance Modeling Using Fields

We induce relevance model #3 (cf., Equation 3) from each field f independently:

$$P(t|f,RM3) = \lambda p(t|q) + (1-\lambda) \sum_{d \in D_{QL}} P(t|f,d) \frac{P(q|f,d)}{\sum_{d' \in D_{QL}} P(q|f,d')} \qquad (4)$$

where $P(t|f,d)$, the probability assigned to term t by a language model induced from field f in document d, is estimated as explained below, and $P(q|f,d) = \prod_{t \in q} P(t|f,d)$.

To estimate $P(t|f,d)$, we should account for the fact that fields are short, and hence, the sparsity problem is exacerbated. For example, the `title` and `heading` fields are usually much shorter than the document `body`. For part B of TREC's ClueWeb09 collection, the average length of `title`, `heading`, and `body` are 7.22, 27.94, and 702.19 terms respectively. Thus, we use a double smoothing approach, where the maximum likelihood estimate (MLE) with respect to a field is Dirichlet smoothed with a linear combination (Jelinek-Mercer) of field-specific and non-field-specific collection MLEs:

$$P(t|f,d) = \frac{c_{t,f,d} + \mu(\beta \frac{c_{t,f}}{|C_f|} + (1-\beta)\frac{c_t}{|C|})}{|d_f| + \mu}; \qquad (5)$$

$c_{t,f,d}$, $c_{t,f}$ and c_t are the counts of t in field f of d, in all fields f in the corpus documents, and in all fields in the corpus; d_f is the length of f in d; $|C_f|$ is the sum of lengths of fields f in all documents; and, $|C|$ is the number of term occurrences in the corpus; β and μ are free parameters.

To score document d with respect to query q, we interpolate the minus cross entropy scores of applying the field-based relevance models from Equation 4 independently on each field:

$$S(d,q) = \sum_{f} W_f \sum_{t \in q} P(t|f,RM3) \log P(t|f,d). \qquad (6)$$

The field weights, W_f, are set using cross-validation.

3.2 Scoring Fields With RM

The method presented above is based on scoring a field using a relevance model induced from the field. Still, fields are short and hence, the induced relevance models might not be robust. Hence, we consider a method which uses a relevance model induced from the entire document ($P(\cdot|RM3)$ from Equation 3) to score each of the document fields. Then, as in Equation 6, the scores are linearly interpolated:

$$S(d,q) = \sum_{f} W_f \sum_{t \in q} P(t|RM3) \log P(t|f,d). \qquad (7)$$

4 EXPERIMENTS

Collections and Fields. Our experiments are ran on the ClueWeb09 Category B collection which contains around 50 million English web pages. We use Indri[1] 5.12 for indexing, and apply the Krovetz stemmer to both documents and queries. Note that stopwords[2] are removed from the query only as stopwords in the documents can have an important influence on the relevance models being induced.

We investigated three fields – `title`, `heading` and `body`. Although `inlink` is a field commonly used in other studies, we omit results for `inlink`, as our preliminary results show that including `inlink` data in the collection can have unexpected consequences. More specifically, Indri and several other systems append `inlink` data from other documents into the linked document, which can change the statistical properties of a document with many `inlinks` significantly. Our experiments show that this destabilizes the relevance models being induced. We leave this unexpected finding to future work as it is an orthogonal problem to the one we wish to explore in this paper. Note that, in our experiments, `heading` is part of `body` as it is an aggregation of the `H1`, `H2`, `H3` and `H4` HTML tags

[1] https://www.lemurproject.org/indri.php

[2] http://www.lemurproject.org/stopwords/stoplist.dft

which are inside the body tag. We believe it might also contain useful information that can be exploited independently of the body.

Retrieval Methods. We used three types of existing retrieval frameworks as baselines: (1) query likelihood (**QL**) and a weighted linear combination of query likelihood over fields (**QLLF**); see Eq. 1 (ii) **BM25** and **BM25F**; and (iii) **SDM** and **SDMF**. For QL, the Dirichlet smoothing parameter μ is set to 2500. For QLLF, μ is 10, 100, 2500 for title, heading, and body and the field weights are 0.2, 0.1, 0.7 respectively. We implemented BM25F [13] (Section 4.1 of the original paper) in Indri, and followed their approach to optimize the parameter weights for {title, heading, body}. The weights were obtained by averaging across a 5-fold cross validation. First we optimized B_f for each field independently. $K1$ was then optimized using B_f from the previous step. Finally, $W_{body} = 1$ was fixed, and W_{title} and $W_{heading}$ were swept. The final parameter choices for ClueWeb09B were $K_1 = 1.02$, title ($B_f = 0.36$, $W_f = 9{,}2$), body ($B_f = 0.32$, $W_f = 2$), and heading ($B_f = 0.16$, $W_f = 1$). For SDMF, we used the configuration from Mohammad et al. [8]: the title, heading and body weights were 0.2, 0.05 and 0.75, respectively. Both ordered and unordered bigram features are only applied over the body field, each of which has a weight of 0.1, whereas the unigram feature of body has a 0.8 weight. Post-hoc spam filtering[3] is applied to all runs with a threshold of 50. Finally, we retain the top 1,000 documents for each ranked list for evaluation. Note that this would affect direct MAP or NDCG comparisons with previous TREC Web Track runs as these were scored over the top 10,000 documents.

Comparison Methodology. An initial list was retrieved using query likelihood with Dirichlet smoothing and $\mu = 2500$. For relevance modeling, we adopted the reranking approach of Diaz [2] who showed that reranking is as effective as retrieval over the entire collection for RM3. As in Equation 4, $P(q|f,d)$ was used instead of $P(q|d)$ for relevance modeling. Thus, the document list was reranked by title, heading, and body query likelihood scores, and the top 50 scored fields were used for relevance modeling independently. Before reranking, we clipped the relevance models for 25 or 50 terms and re-normalized term weights. Then the 1,000 documents were reranked with RM3 over fields for the three resulting lists. Document scores from the three ranked lists were then linearly combined to produce the final ranking. A ten-fold cross validation was performed for tuning the relevance model clipping (number of terms) and RM3 query weights.

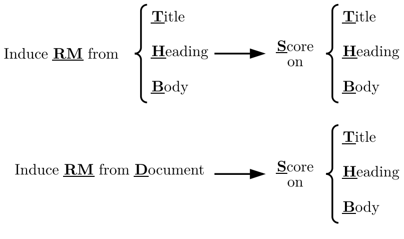


Figure 1: Experimental naming rules.

Naming followed the rules outlined in Figure 1. **RMTST** is a relevance model induced from the titles of a rank list, and used to rerank the list by scoring the title field. A linear combination of **RMTST**, **RMHSH**, and **RMBSB** is named as **RMFLF**. If we induce relevance models from documents, and rerank documents by title, we name it **RMDST**. Finally, the linear combination of scores of **RMDST**, **RMDSH**, and **RMDSB** is called **RMDLF**.

Name	MAP	P@5	NDCG@20
BM25	0.196	0.359	0.234
BM25F	0.203	0.401†	0.256†
SDM	0.210	0.366	0.253
SDMF	0.200	0.398†	0.256
QL	0.196	0.347	0.238
QLLF	0.203	0.398†	0.256†
RM3	0.205	0.378	0.244
RMFLF	0.198	0.420†	0.257
RMDLF	0.197	0.420†	0.252

Table 1: Effectiveness of field-based retrieval methods. A pairwise, two-tailed t-test was performed between a non-field model (BM25,SDM,QL,RM3) and the corresponding extended field-based model. A † denotes significance at $p \leq 0.05$.

Field-based models for document retrieval. First we consider the impact of field information on the retrieval effectiveness in Table 1. We can observe a consistent trend for all retrieval methods without relevance modeling: field information improves early precision. For P@5 and NDCG@20, statistically significant differences are observed, with the exception of SDM and SDMF. Although slight MAP improvements are observed in Table 1, the results are not significant.

Next, we consider the relevance modeling based methods: RM3, RMFLF and RMDLF, which follow a similar trend as the methods without relevance modeling: early precision benefits from incorporating field information, but the improvements are not statistically significant for NDCG@20 or MAP. The early precision of the relevance model approaches is also better than that of the baselines. In general, the results shown in Table 1 suggest that fields in documents can provide new relevance signals to some extent, and integrating the fields into existing retrieval models improves retrieval effectiveness; but the improvements are somewhat volatile depending on what is being measured.

Field-Based Retrieval and Relevance Modelling. Finally, we explore if the field-based retrieval methods can be further improved using relevance modeling techniques. In order to gain a better understanding, we conducted experiments using two different settings: (a) a PRF setting, where pseudo-relevance feedback documents are used; and (b) an oracle setting, in which the first five relevant (by QREL) documents from the initial list are used. As we showed in Table 1, using a linearly combined field-based relevance models improves early precision significantly, and we further analyze the effectiveness for each field independently now.

As described in Section 3, we induce relevance models using different sources: (i) the fields themselves; and (ii) the entire document. We consider the methods from the RMFLF and RMDLF family. The trends for both categories are shown in Table 2(a). The body field is more effective than either heading or title fields, the title is slightly

[3] https://plg.uwaterloo.ca/ gvcormac/clueweb09spam/

	Field	Wt	MAP	P@5	NDCG@20
RM3	-	-	0.205	0.378	0.244
RMFLF	RMTST	0.1	0.122†	0.265†	0.151†
	RMHSH	0.1	0.110†	0.246†	0.137†
	RMBSB	0.8	0.189†	0.391	0.234
	Linear	-	0.198	0.420†	0.257
RMDLF	RMDST	0.2	0.130†	0.277†	0.159†
	RMDSH	0.1	0.104†	0.248†	0.136†
	RMDSB	0.7	0.187†	0.380	0.231†
	Linear	-	0.197	0.420†	0.252

(a) PRF settings

	Field	Wt	MAP	P@5	NDCG@20
RM3	-	-	0.298	0.678	0.445
RMFLF	RMTST	0.2	0.256†	0.646	0.432
	RMHSH	0.1	0.206†	0.618†	0.393†
	RMBSB	0.7	0.277†	0.674	0.438
	Linear	-	0.293	0.732†	0.479†
RMDLF	RMDST	0.1	0.152†	0.296†	0.200†
	RMDSH	0.1	0.126†	0.308†	0.174†
	RMDSB	0.8	0.266†	0.640†	0.406†
	Linear	-	0.276†	0.656	0.417†

(b) Oracle settings

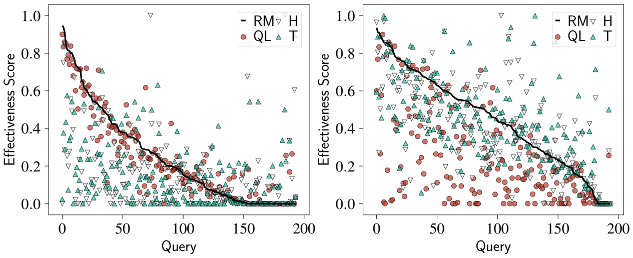
Table 2: Decomposition of field-based relevance modeling for both (a) PRF and the (b) oracle settings. A pairwise, two-tailed t-test was performed between each model and RM3. A † denotes significance at $p \leq 0.05$.

more effective than the `heading` field, and a linear combination of all three methods provides the greatest effectiveness improvement.

When constructing relevance models based on the entire document and applying the constructed model to score each field (RMDLF), there are some differences from RMFLF, but the interpolated scores (Linear) perform very similar, and not significantly different.

The effectiveness of using true relevant documents to construct relevance models is shown in Table 2(b). This set of experiments reveals the potential effectiveness gains we might achieve using relevance modeling over fields. We observe that, if we apply a relevance model induced from the entire document to each field, system effectiveness is degraded. This confirms the observation made in the PRF experiment. More importantly, RMFLF outperforms RMDLF which suggests that inducing relevance models from fields instead of documents is a promising approach that we do not yet understand how to exploit. Experimental results of the oracle settings confirm that applying relevance modeling techniques can significantly improve the performance of field-based retrieval methods, particularly for early precision metrics.

Per-Query Performance Breakdown. In order to better understand the performance patterns, we performed a failure analysis for RMFLF on a per query basis, as shown in Figure 2. In both instances we can see that the two field based methods, RMTST and RMHSH, improve performance on tail queries where RM3 and QL have low NDCG@20 scores. When considering the PRF setting and the 25% worst-performing queries for RM3, 58% and 44% can be improved by using RMTST and RMHSH, respectively. In an oracle setting, 44% and 38% of queries among the worst 25% RM3 tail queries are improved. However, none of the current field-based relevance models are robust for all queries, and the performance can be worse than either QL or non-field-based standard RM3.



(a) PRF settings

(b) Oracle settings

Figure 2: Performance breakdown for RMFLF methods on a per-query basis. The evaluation metric is NDCG@20 and all topics are organized w.r.t. the RM3 methods. The "RM" means RM3, the "H" means RMHSH and the "T" means RMTST method.

5 CONCLUSIONS

In this paper, we first examined three techniques for using field information in semi-structured documents. They all outperform their non-field counterparts in terms of early precision. We then incorporated field information into relevance modeling, and observed similar trends – early precision is significantly improved. Our oracle experiments suggest that fields could be an important source of information that might be further exploited in relevance modeling. The most interesting finding is that difficult queries are improved using field-based relevance modeling, which is a promising direction for future study.

Acknowledgements. This work was supported by the Australian Research Council's *Discovery Projects* Scheme (DP170102231).

REFERENCES

[1] N. Abdul-Jaleel, J. Allan, W. B. Croft, F. Diaz, L. Larkey, X. Li, M. D., and C. Wade. 2004. UMASS at TREC 2004 — Novelty and HARD.

[2] F. Diaz. 2015. Condensed List Relevance Models. In *Proc. ICTIR*. 313–316.

[3] L. Gallagher, R. Chen, J. Mackenzie, F. Scholer, R. Benham, and J. S. Culpepper. 2013. RMIT at the NTCIR-13 We Want Web Task. In *Proc. NTCIR*.

[4] J. Y. Kim and W. B. Croft. 2012. A Field Relevance Model for Structured Document Retrieval. In *Proc. ECIR*. 97–108.

[5] V. Lavrenko and W. B. Croft. 2001. Relevance Based Language Models. In *Proc. SIGIR*. 120–127.

[6] D. Metzler and W. B. Croft. 2005. A Markov Random Field Model for Term Dependencies. In *Proc. SIGIR*. 472–479.

[7] D. Metzler, T. Strohman, Y. Zhou, and W. B. Croft. 2005. Indri at TREC 2005: Terabyte Track. In *Proc. TREC*.

[8] H. R. Mohammad, K. Xu, J. Callan, and J. S. Culpepper. 2018. Dynamic Shard Cutoff Prediction for Selective Search. In *Proc. SIGIR*. 85–94.

[9] P. Ogilvie and J. P. Callan. 2003. Combining document representations for known-item search. In *Proc. SIGIR*. 143–150.

[10] S. Robertson, S. Walker, S. Jones, M. M. Hancock-Beaulieu, M. Gatford, and others. 1994. Okapi at TREC-3. In *Proc. TREC*. 109–126.

[11] S. Robertson, H. Zaragoza, and M. Taylor. 2004. Simple BM25 Extension to Multiple Weighted Fields. In *Proc. CIKM*. 42–49.

[12] H. Zamani, B. Mitra, X. Song, N. Craswell, and S. Tiwary. 2018. Neural Ranking Models with Multiple Document Fields. In *Proc. WSDM*. 700–708.

[13] H. Zaragoza, N. Craswell, M. Taylor, S. Saria, and S. Robertson. 2004. Microsoft Cambridge at TREC 13: Web and Hard Tracks. In *Proc. TREC*.