Position-Aware Representations for Relevance Matching in Neural Information Retrieval

Kai Hui
Saarbrücken Graduate School of Computer Science/ Max Planck Institute for Informatics
khui@mpi-inf.mpg.de

Andrew Yates
Max Planck Institute for Informatics
ayates@mpi-inf.mpg.de

Klaus Berberich
Max Planck Institute for Informatics/ htw saar
kberberi@mpi-inf.mpg.de

Gerard de Melo
Rutgers University
New Brunswick, NJ
gdm@demelo.org

ABSTRACT

To deploy deep learning models for ad-hoc information retrieval, suitable representations of query-document pairs are needed. Such representations ought to capture all relevant information required to assess the relevance of a document for a given query, including uni-gram term overlap as well as positional information such as proximity and term dependencies. In this work, we investigate the use of similarity matrices that are able to encode such position-specific information. Extensive experiments on TREC Web Track data confirm that such representations can yield good results.

1. INTRODUCTION

Given the success of deep learning, it is important to study to what extent it can benefit ad-hoc information retrieval. A challenge here has been how to induce suitable representations of query-document pairs, based on which one can learn a model for relevance assessments. Such representations need to capture information pertaining to both queries and documents, and, in particular, their relationships and interactions. Early work attempted to represent queries and documents separately [5], neglecting valuable interaction information such as unigram matches. Inspired by traditional retrieval models, Guo et al. [1] pointed out the benefits of representations that account for local matching signals and also preserve exact matching information, i.e., the query term occurrences. Accordingly, they opted to rely on similarity histograms, discarding positional information, which they deemed less important. In extensive experimental comparisons, the proposed DRMM [1] outperformed all baseline deep retrieval models, and, more importantly, DRMM proved to be the only deep model able to outperform traditional baselines, such as the query likelihood model, on standard TREC benchmarks.

However, beyond locality and exact match information, established retrieval models have also shown the importance of position-related features. Among them, for example, term dependency [2] and proximity [6] of query term occurrences both contribute significantly to the quality of search results. In this paper, we thus investigate how to overcome this important shortcoming of previous work. In the approach by Guo et al. [1], each query term is associated with a histogram indicating how well the terms from a document match with it at different similarity levels. As mentioned, however, such histograms discard all positional information, making it impossible for the remaining network architecture to learn from position-related cues. As in [4], we investigate a notion of similarity matrices specifically designed to preserve such signals. For a fair comparison under equal conditions, we then feed these representations into the same neural network architecture as used by [1], changing only the representations. Based on this, the network then assesses the relevance of a document given the query. Note, however, that different representations may also affect the architecture of the deep model. For example, with a similarity matrix, a locally connected network such as a convolutional neural network may be more suitable to learn n-gram matches. To cope with this, we thus investigate two variants for extracting a similarity matrix from query-document pairs, and empirically compare both methods against the similarity histograms of [1] on widely used TREC Web Track benchmarks. Our extensive experiments demonstrate that our similarity matrix approach can lead to significantly better retrieval results. Overall, our contributions are twofold: 1) we highlight and demonstrate the importance of preserving positional information in representations for neural IR; and 2) we present a novel method to extract similarity matrices and confirm their empirical effectiveness on standard benchmarks.

2. METHODOLOGY

We now describe our two methods for extracting a similarity matrix from a given query-document pair. We refer to the DRMM model with regular similarity histograms [1] as *DRMM-histogram* and to the two variants as *DRMM-matrix-max* and *DRMM-matrix-firstk*, respectively.

Similarity matrix. Given a document d and a query q, the similarity between every term pair from d and q can be encoded in a similarity matrix $sim_{|q|\times|d|}$, where sim_{ij} quantifies the degree of similarity between the i-th term from the query q and the j-th term from the document d. As in [1], the cosine similarity between every pair of terms is

WWW'17 Companion, April 3–7, 2017, Perth, Australia.
ACM 978-1-4503-4914-7/17/04.
http://dx.doi.org/10.1145/3041021.3054258

	year	DRMM-matrix-firstk		DRMM-matrix-max		DRMM-histogram		QL	
		ERR@20	nDCG@20	ERR@20	nDCG@20	ERR@20	nDCG@20	ERR@20	nDCG@20
ALL	wt12	0.264 ↑	0.190 ↑	0.235 ↑	0.171 ↑	0.265 ↑	0.169 ↑	0.177	0.106
	wt13	0.116	0.244 ↑	0.132 ↑	0.245 ↑	0.115	0.197	0.101	0.190
	wt14	0.139	0.212	0.190 ↑	0.278 ↑	0.154	0.243	0.131	0.231
NOSPAM	wt12	0.235 ↑	0.184 ↑	0.255 ↑	0.166 ↑	0.230	0.182 ↑	0.190	0.132
	wt13	0.131 ↑	0.238 ↑⇑	0.138 ↑	0.243 ↑⇑	0.125	0.189	0.095	0.180
	wt14	0.140	0.201 ↓	0.166	0.258	0.144	0.224	0.159	0.261

Table 1: ERR@20 and nDCG@20 on TREC Web Track 2012–14 when re-ranking search results from QL. ↑ and ⇑ (↓ and ⇓) indicate significantly better (worse) results relative to QL baseline and *DRMM-histogram* respectively. Two-tailed paired Student's t-test and 95% confidence intervals are employed.

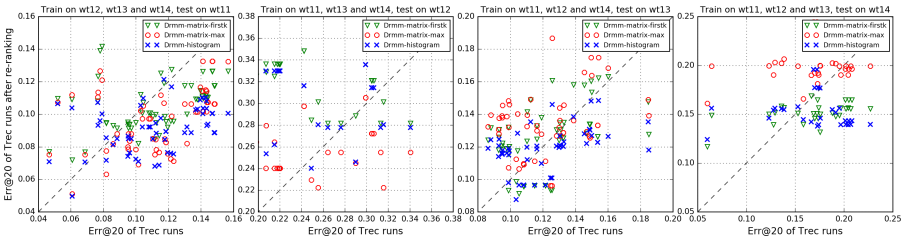

Figure 1: Re-ranking search results from Trec Web Track 2011–14 runs. The x-axis represents the ERR@20 of the original TREC runs and the y-axis represents the ERR@20 of the corresponding re-ranked runs. The dashed line is y=x.

calculated via their *word2vec* vectors [3], using the publicly available pre-trained Google News embeddings[1].

Distilling the similarity matrix. There are two principal reasons to further distill the similarity matrices: 1) DRMM [1], and also many convolutional neural architectures, expect a uniform dimensionality across all inputs, i.e. for different query-document pairs. 2) More importantly, it is preferable for the relevance matching to be captured locally instead of being distributed over a long series of separate matrix entries for an entire document. Hence, we select the most significant parts of a document's representation for the relevance assessment. Along the query term dimension, we zero-pad the similarity matrices resulting from different queries to the maximum length among all queries. As for the document term dimension, we propose selecting k columns to retain in the similarity matrix. To this end, *DRMM-matrix-firstk*, as in [4], keeps the first k columns in the matrix. We further propose *DRMM-matrix-max*, which retains the top-k positions with the highest similarity relative to any query terms. For both variants, matrices for any documents shorter than length k are zero-padded.

3. EVALUATION

In this section, we empirically compare *DRMM-matrix-firstk* and *DRMM-matrix-max* with *DRMM-histogram*, which, as mentioned, is the state-of-the-art deep retrieval model. We rely on the 2011–2014 TREC Web Track benchmarks[2]. In total, this data consists of 200 queries, 64k judgments, and the runs submitted by participants of the TREC Web Track (62 runs for 2011, 48 for 2012, 50 for 2013, and 27 for 2014). The DRMM model proposed in [1] is used to train and test all models. The judged documents are split such that three years (150 queries total) are used for training, and one year (50 queries) is used for testing. We randomly reserve 30 queries among the training queries for validation, tuning the number of columns k to retain. As in [1], we fix the length of the similarity histograms to 30 in *DRMM-histogram*. We first examine how well the deep models can re-rank search results from one of the most used baselines, namely query likelihood, coded as QL. Furthermore, we re-rank all runs from TREC and examine whether the deep model delivers improved results when re-ranking search results from different runs. The 2012–14 QL baselines on ClueWeb A from the TREC Web Track[3] are used: search results filtering out spam pages are coded as NOSPAM, while those without spam pages filtering is coded as ALL. The results are summarized in terms of ERR@20 and nDCG@20 in Table 1. Similarly, in Figure 1, all runs are re-ranked and the results are plotted in terms of ERR@20 before and after the re-ranking. From Table 1, we can infer that with similarity matrices as input, one can improve the QL results significantly, and that *DRMM-matrix-max* is particularly well-suited for such ad-hoc retrieval. When making comparisons among the three DRMM models, one can conclude that the models with similarity matrices as input perform at least as good as the model using histograms.

[1] `https://code.google.com/p/word2vec`
[2] `http://trec.nist.gov/tracks.html`

4. REFERENCES

[1] J. Guo, Y. Fan, Q. Ai, and W. B. Croft. A deep relevance matching model for ad-hoc retrieval. In *CIKM* 2016.

[2] D. Metzler and W. B. Croft. A markov random field model for term dependencies. In *SIGIR* 2005.

[3] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean. Distributed representations of words and phrases and their compositionality. In *NIPS* 2013.

[4] B. Mitra, F. Diaz, and N. Craswell. Learning to match using local and distributed representations of text for web search. *arXiv*, 2016.

[5] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil. A latent semantic model with convolutional-pooling structure for information retrieval. In *CIKM* 2014.

[6] T. Tao and C. Zhai. An exploration of proximity measures in information retrieval. In *SIGIR* 2007.

[3] `https://github.com/trec-web/trec-web-2014`