MPII at the NTCIR-14 WWW-2 Task

Andrew Yates

Max Planck Institute for Informatics, Saarbrücken, Germany ayates@mpi-inf.mpg.de

Abstract. MPII participated in the English subtask of the NTCIR-14 WWW-2 Task. We evaluated several variants of the PACRR neural re-ranking model, considering smaller vs. larger models and the impact of cascade pooling. No significant differences were found between any of the runs.

Team Name. MPII

Subtasks. English

Keywords: neural IR · Web search · ad-hoc retrieval

1 Introduction

The MPII team participated in the English subtask of the NTCIR-14 WWW-2 Task [4] with the goal of evaluating several variants of the PACRR [2] and Co-PACRR [3] neural re-ranking models. Both models consider the embedding similarity between query and document terms at different levels of granularity in order to consider term dependencies and the approximate location of query terms (e.g., near the beginning vs. in the middle of the document), which is inspired by the cascade model [1]. In particular, we submitted five runs with varying hyperparameters in order to evaluate whether these parameters significantly affected the models' performance. The remainder of this report describes our methodology in more detail and discusses the runs' results.

2 Methodology

In this section we briefly describe the PACRR models and the variants used in our runs. Detailed descriptions of the models can be found in their respective papers. [2, 3]

2.1 Base model

PACRR takes a query-document similarity matrix as input that encodes the cosine similarities between the embeddings for each query and document term. This similarity matrix is processed by a CNN with filters of size $n \times n$, with a separate CNN for each n-gram size considered. Max pooling is performed on the filter dimension, followed by k-max pooling on the query dimension. This matching component's output size is $l_q \times n \times k$, where l_q is the query length. The matching component's output is then converted into a query-document relevance score. The original PACRR model uses a LSTM to consume a sequence of $n \times k$ signals for each query term, while Co-PACRR uses a series of fully connected layers to consume the output for all query terms at once. This final component is called the combination layer.

2.2 Variants

Our submitted runs diverged from the base model described in two ways: we used a different combination layer in all five runs, and we included Co-PACRR's cascade k-max pooling in three of the five runs.
Combination layer. Rather than using a fully connected layer or a LSTM to combine per-query term match signals into a document relevance score, we used a series of two fully connected layers to produce a score for each query term. These per-query term scores were summed to produce the final relevance score.
Cascade k-max pooling. We replaced PACRR's k-max pooling layer with Co-PACRR's cascade k-max pooling layer in several runs. This allows the combination layer to consider the rough location of term matches in the document.

3 Evaluation

3.1 Experimental setup

All runs were trained on the documents, queries, and relevance judgments from the TREC 2009-2013 Web Track (following [3]). NTCIR WWW-1 and the 2014 Web Track (WT14) were reserved for validation. The runs were submitted using weights from the best epoch on WWW-1 data as measured by nDCG@20. We re-ranked the BM25 baseline results provided by the WWW-2 organizers, and we did not consider any features based on the document URLs. All submitted runs were trained using hinge loss for 150 iterations consisting of 4096 instances with a batch size of 64. The number of filters for each CNN was set to 16. The maximum n-gram hyperparameter was set to 3, meaning unigrams, bigrams, and trigrams were considered.

The five runs differed in terms of whether cascade k-max pooling was used, the value of k, and the size of the combination component's hidden layer. The values for these hyperparameters are shown in Table 1. Runs 1-3 use cascade k-max pooling to include information about the number of matches in the first 25%, 50%, 75%, and 100% of documents, while runs 4-5 use k-max pooling only. Each of these two groups has one "small" variant and one "large" variant.

3.2 Results

The WWW-2 results from MPII's five runs are shown in Table 2. Run 3, which combined cascade pooling with a low k and high hidden layer size, performed the

Run	Cascade pooling	*k*-max pooling	Hidden layer size
MPII-E-CO-NU-Base-1	25%, 50%, 75%, 100%	5	1
MPII-E-CO-NU-Base-2	25%, 50%, 75%, 100%	15	8
MPII-E-CO-NU-Base-3	25%, 50%, 75%, 100%	5	8
MPII-E-CO-NU-Base-4	100%	5	1
MPII-E-CO-NU-Base-5	100%	15	8

Table 1. Model hyperparameters that varied across runs. *Cascade pooling* indicates the document positions considered, *k-max pooling* indicates the number of term matches considered for each query term, and *hidden layer size* indicates the size of the hidden layer used in the combination component.

Run	nDCG@10	Q@10	nERR@10
MPII-E-CO-NU-Base-1	0.3204	0.3009	0.4541
MPII-E-CO-NU-Base-2	0.3394	0.3255	0.4590
MPII-E-CO-NU-Base-3	**0.3413**	0.3183	0.4658
MPII-E-CO-NU-Base-4	0.3336	**0.3265**	**0.4723**
MPII-E-CO-NU-Base-5	0.3293	0.3110	0.4584

Table 2. Results from NTCIR-14 WWW-2. According to the overview paper [4], there are no significant differences between run pairs.

best in terms of nDCG. Run 4, which used a small hidden layer size and did not use cascade pooling, performed the best in terms of Q and nERR. There are no significant differences between any pair of runs, however, making it impossible to draw conclusions about the effects of these hyperparameters.

4 Conclusions

In this work we described MPII's participation in the NTCIR-14 WWW-2 task. MPII submitted five runs based on variants of the PACRR and Co-PACRR neural re-ranking models. While the runs performed slightly differently in terms of the metrics considered (e.g., up to a 6.5% improvement in nDCG@10), there were no significant differences between them.

5 Acknowledgements

We gratefully acknowledge the support of NVIDIA Corporation with the donation of a Titan X Pascal GPU used in this research.

References

1. Craswell, N., Zoeter, O., Taylor, M., Ramsey, B.: An experimental comparison of click position-bias models. In: Proceedings of the 2008 International Conference on Web Search and Data Mining. pp. 87–94. WSDM '08,

ACM, New York, NY, USA (2008). https://doi.org/10.1145/1341531.1341545, http://doi.acm.org/10.1145/1341531.1341545

2. Hui, K., Yates, A., Berberich, K., de Melo, G.: PACRR: A position-aware neural IR model for relevance matching. In: EMNLP. pp. 1049–1058. Association for Computational Linguistics (2017)
3. Hui, K., Yates, A., Berberich, K., de Melo, G.: Co-pacrr: A context-aware neural IR model for ad-hoc retrieval. In: WSDM. pp. 279–287. ACM (2018)
4. Mao, J., Sakai, T., Luo, C., Xiao, P., Liu, Y., Dou, Z.: Overview of the NTCIR-14 We Want Web task. In: Proceedings of the 14th NTCIR Conference on Evaluation of Information Access Technologies (2019)