AN UPDATED DUET MODEL FOR PASSAGE RE-RANKING

Bhaskar Mitra
Microsoft AI & Research
Montreal, Canada
bmitra@microsoft.com

Nick Craswell
Microsoft AI & Research
Redmond, USA
nickcr@microsoft.com

ABSTRACT

We propose several small modifications to Duet—a deep neural ranking model—and evaluate the updated model on the MS MARCO passage ranking task. We report significant improvements from the proposed changes based on an ablation study.

消融

Keywords Neural information retrieval · Passage ranking · Ad-hoc retrieval · Deep learning

1 Introduction

In information retrieval (IR), traditional learning to rank [Liu, 2009] models estimate the relevance of a document to a query based on hand-engineered features. The input to these models typically includes, among others, features based on patterns of exact matches of query terms in the document. Recently proposed deep neural IR models [Mitra and Craswell, 2018], in contrast, accept the raw query and document text as input. The input text is represented as one-hot encoding of words (or sub-word components [Kim et al., 2016, Jozefowicz et al., 2016, Sennrich et al., 2015])—and the deep neural models focus primarily on learning latent representations of text that are effective for matching query and document. Mitra et al. [2017] posit that deep neural ranking models should focus on both: (i) representation learning for text matching, as well as on (ii) feature learning based on patterns of exact matches of query terms in the document. They demonstrate that a neural ranking model called *Duet*[1]—with two distinct sub-models that consider both matches in the term space (the *local sub-model*) and the learned latent space (the *distributed sub-model*)—is more effective at estimating query-document relevance.

In this work, we evaluate a duet model on the MS MARCO passage ranking task [Bajaj et al., 2016]. We propose several simple modifications to the original Duet architecture and demonstrate through an ablation study that incorporating these changes results in significant improvements on the passage ranking task.

2 Passage re-ranking on MS MARCO

The MS MARCO passage ranking task [Bajaj et al., 2016] requires a model to rank approximately thousand passages for each query. The queries are sampled from Bing's search logs, and then manually annotated to restrict them to questions with specific answers. A BM25 [Robertson et al., 2009] model is employed to retrieve the top thousand candidate passages for each query from the collection. For each query, zero or more candidate passages are deemed relevant based on manual annotations. The ranking model is evaluated on this passage re-ranking task using the mean reciprocal rank (MRR) metric [Craswell, 2009]. Participants are required to submit the ranked list of passages per query for a development (dev) set and a heldout (eval) set. The ground truth annotations for the development set are available publicly, while the corresponding annotations for the evaluation set are heldout to avoid overfitting. A public leaderboard[2] presents all submitted runs from different participants on this task.

[1] While Mitra et al. [2017] propose a specific neural architecture, they refer more broadly to the family of neural architectures that operate on both term space and learned latent space as duet. We refer to the specific architecture proposed by Mitra et al. [2017] as Duet—to distinguish it from the general family of such architectures that we refer to as duet (note the difference in capitilization).

[2] http://www.msmarco.org/leaders.aspx

3 The updated Duet model

local sub_model

distributed sub-model

In this section, we briefly describe several modifications to the Duet model. A public implementation of the updated Duet model using PyTorch [Paszke et al., 2017] is available online[3].

Word embeddings We replace the character level n-graph encoding in the input of the distributed model with word embeddings. We see significant reduction in training time given a fixed number of minibatches and a fixed minibatch size. This change primarily helps us to train on a significantly larger amount of data under fixed training time constraints. We initialize the word embeddings using pre-trained GloVe [Pennington et al., 2014] embeddings before training the Duet model.

输入是预训练好的 GloVe

Inverse document frequency weighting In contrast to some of the other datasets on which the Duet model has been previously evaluated [Mitra et al., 2017, Nanni et al., 2017], the MS MARCO dataset contains a relatively larger percentage of natural language queries and the queries are considerably longer on average. In traditional IR models, the inverse document frequency (IDF) [Robertson, 2004] of a query term provides an effective mechanism for weighting the query terms by their discriminative power. In the original Duet model, the input to the local sub-model corresponding to a query q and a document d is a binary interaction matrix $X \in \mathbb{R}^{|q|\times|d|}$ defined as follows:

区别 query term 的权重

原始 Duet model: local sub-model 的输入: 一个二进制指示矩阵 $|q|\times|d|$

$$X_{ij} = \begin{cases} 1, & \text{if } q_i = d_j \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

We incorporate IDF in the Duet model by weighting the interaction matrix by the IDF of the matched terms. We adopt the Robertson-Walker definition of IDF [Jones et al., 2000] normalized to the range $[0, 1]$.

改进后 local sub-model 的输入: 融合进 IDF 值. 矩阵维度 $|q|\times|d|$

X'_{ij}

$$X'_{ij} = \begin{cases} \text{IDF}(q_i), & \text{if } q_i = d_j \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

$$\text{IDF}(t) = \frac{\log(N/n_t)}{\log(N)} \tag{3}$$

Where, N is the total number of passages in the collection and n_t is the number of passages in which the term t appears at least once.

原始 Duet model: local sub-model 和 distributed sub-model 各产生一个分值, 然后加权得到最终相似度分值.

Non-linear combination of local and distributed models [Zamani et al., 2018] show that when combining different sub-models in a neural ranking model, it is more effective if each sub-model produce a vector output that are further combined by additional multi-layer perceptrons (MLP). In the original Duet model, the local and the distributed sub-models produce a single score that are linearly combined. In our updated architecture, both models produce a vector that are further combined by an MLP—with two hidden layers—to generate the estimated relevance score.

local 和 distributed sub-model 各产生一个向量, 然后通过 2层 MLP 产生最终分值.

Rectifier Linear Units (ReLU) We replace the Tanh non-linearities in the original Duet model with ReLU [Glorot et al., 2011] activations.

tanh ⟹ ReLU

Bagging We observe some additional improvements from combining multiple Duet models—trained with different random seeds and on different random sample of the training data—using bagging [Breiman, 1996].

采用 Bagging (不同的随机种子和训练数据)

4 Experiments

The MS MARCO task provides a pre-processed training dataset—called "triples.train.full.tsv"—where each training sample consists of a triple $\langle q, p_+, p_-\rangle$, where q is a query and p_+ and p_- are a pair of passages, with p_+ being more relevant to q than p_-. Similar to the original Duet model, we employ the cross-entropy with softmax loss to learn the parameters of our model $\mathcal{M}$:

交叉熵损失函数

[3] https://github.com/dfcf93/MSMARCO/blob/master/Ranking/Baselines/Duet.ipynb

Table 1: Comparison of the different Duet variants and other state-of-the-art approaches from the public MS MARCO leaderboard. The update Duet model—referred to as Duet v2—benefits significantly from the modifications proposed in this paper.

Model	MRR@10	
	Dev	Eval
Other approaches		
BM25	0.165	0.167
Single CKNRM [Dai et al., 2018] model	0.247	0.247
Ensemble of 8 CKNRM [Dai et al., 2018] models	0.290	0.271
IRNet (a proprietary deep neural model)	0.278	0.281
BERT [Nogueira and Cho, 2019]	0.365	0.359
Duet variants		
Single Duet v2 w/o IDF weighting for interaction matrix	0.163	-
Single Duet v2 w/ Tanh non-linearity (instead of ReLU)	0.179	-
Single Duet v2 w/o MLP to combine local and distributed scores	0.208	-
Single Duet v2 model	0.243	0.245
Ensemble of 8 Duet v2 models	0.252	0.253

加入IDF很重要

单个 Duet model

8个 Duet model 集成

$$\mathcal{L} = \mathbb{E}_{q,p_+,p_- \sim \theta}[\ell(\mathcal{M}_{q,p_+} - \mathcal{M}_{q,p_-})] \tag{4}$$

$$\text{where, } \ell(\Delta) = \log(1 + e^{-\sigma \cdot \Delta}) \tag{5}$$

Where, $\mathcal{M}_{q,p}$ is the relevance score for the pair $\langle q, p \rangle$ as estimated by the model $\mathcal{M}$. Note, that by considering a single negative passage per sample, our loss is equivalent to the RankNet loss [Burges et al., 2005].

训练的超参设置

We use the Adam optimizer with default parameters and a learning rate of 0.001. We set σ in Equation 5 to 0.1 and dropout rate for the model to 0.5. We trim all queries and passages to their first 20 and 200 words, respectively. We restrict our input vocabulary to the 71,486 most frequent terms in the collection and set the size of all hidden layers to 300. We use minibatches of size 1024 and train the model for 1024 minibatches. Finally, for bagging we train eight different Duet models with different random seeds and on different samples of the training data. We train and evaluate our models using a Tesla K40 GPU—on which it takes a total of only 1.5 hours to train each single Duet model and to evaluate it on both dev and eval sets.

5 Results

Table 1 presents the MRR@10 corresponding to all the Duet variants we evaluated on the dev set. The updated Duet model with all the modifications described in Section 3—referred hereafter as Duet v2—achieves an MRR@10 of 0.243. We perform an ablation study by leaving out one of the three modifications—(i) IDF weighting for interaction matrix, (ii) ReLU non-linearity instead of Tanh, and (iii) LP to combine local and distributed scores,—out at a time. We observe a 33% degradation in MRR by not incorporating the IDF weighting alone. It is interesting to note that the Github implementations[4] of the KNRM [Xiong et al., 2017] and CKNRM [Dai et al., 2018] models also indicate that their MS MARCO submissions incorporated IDF term-weighting—potentially indicating the value of IDF weighting across multiple architectures. Similarly, we also observe a 26% degradation in MRR by using Tanh non-linearity instead of ReLU. Using a linear combination of scores from the local and the distributed model instead of combining their vector outputs using an MLP results in 14% degradation in MRR. Finally, we observe a 3% improvement in MRR by ensembling eight Duet v2 models using bagging. We also submit the individual Duet v2 model and the ensemble of eight Duet v2 models for evaluation on the heldout set and observe similar numbers.

We include the MRR numbers for other non-Duet based approaches that are available on the public leaderboard in Table 1. As of writing this paper, BERT [Devlin et al., 2018] based approaches—*e.g.*, [Nogueira and Cho, 2019]—are outperforming other approaches by a significant margin. Among the non-BERT based approaches, a proprietary deep neural model—called IRNet—currently demonstrates the best performance on the heldout evaluation set. This is followed, among others, by an ensemble of CKNRM [Dai et al., 2018] models and the single CKNRM model. The single Duet v2 model achieves comparable MRR to the single CKNRM model on the eval set. The ensemble of Duet v2 models, however, performs slightly worse than the ensemble of the CKNRM models on the same set.

[4] https://github.com/thunlp/Kernel-Based-Neural-Ranking-Models

6 Discussion and conclusion

In this paper, we describe several simple modifications to the original Duet model that result in significant improvements over the original architecture on the MS MARCO task. The updated architecture—we call Duet v2—achieves comparable performance to other non-BERT based top performing approaches, as listed on the public MS MARCO leaderboard. We note, that the Duet v2 model we evaluate contains significantly fewer learnable parameters—approximately 33 million—compared to other top performing approaches, such as BERT based models [Nogueira and Cho, 2019] and single CKNRM model [Dai et al., 2018]—both of which contains few hundred million learnable parameters. Comparing the models based on the exact number of learnable parameters, however, may not be meaningful as most of these parameters are due to large vocabulary size in the input embedding layers. It is not clear how significantly the vocabulary size impacts model performance—an aspect we may want to analyse in the future. It is worth emphasizing that compared to other top performing approaches, training the Duet v2 model takes significantly less resource and time—1.5 hours to train a single Duet model and to evaluate it on both dev and eval sets using a Tesla K40 GPU—which may make the model an attractive starting point for new MS MARCO participants. The model performance on the MS MARCO task may be further improved by adding more depth and / or more careful hyperparameter tuning.

References

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*, 2016.

Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

Chris Burges, Tal Shaked, Erin Renshaw, Ari Lazier, Matt Deeds, Nicole Hamilton, and Greg Hullender. Learning to rank using gradient descent. In *Proceedings of the 22nd international conference on Machine learning*, pages 89–96. ACM, 2005.

Nick Craswell. Mean reciprocal rank. In *Encyclopedia of Database Systems*, pages 1703–1703. Springer, 2009.

Zhuyun Dai, Chenyan Xiong, Jamie Callan, and Zhiyuan Liu. Convolutional neural networks for soft-matching n-grams in ad-hoc search. In *Proceedings of the eleventh ACM international conference on web search and data mining*, pages 126–134. ACM, 2018.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 315–323, 2011.

K Sparck Jones, Steve Walker, and Stephen E. Robertson. A probabilistic model of information retrieval: development and comparative experiments: Part 2. *Information processing & management*, 36(6):809–840, 2000.

Rafal Jozefowicz, Oriol Vinyals, Mike Schuster, Noam Shazeer, and Yonghui Wu. Exploring the limits of language modeling. *arXiv preprint arXiv:1602.02410*, 2016.

Yoon Kim, Yacine Jernite, David Sontag, and Alexander M Rush. Character-aware neural language models. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

Tie-Yan Liu. Learning to rank for information retrieval. *Foundation and Trends in Information Retrieval*, 3(3):225–331, March 2009.

Bhaskar Mitra and Nick Craswell. An introduction to neural information retrieval. *Foundations and Trends® in Information Retrieval (to appear)*, 2018.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. Learning to match using local and distributed representations of text for web search. In *Proc. WWW*, pages 1291–1299, 2017.

Federico Nanni, Bhaskar Mitra, Matt Magnusson, and Laura Dietz. Benchmark for complex answer retrieval. In *Proc. ICTIR*. ACM, 2017.

Rodrigo Nogueira and Kyunghyun Cho. Passage re-ranking with bert. *arXiv preprint arXiv:1901.04085*, 2019.

Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in pytorch. 2017.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. *Proc. EMNLP*, 12:1532–1543, 2014.

Stephen Robertson. Understanding inverse document frequency: on theoretical arguments for idf. *Journal of documentation*, 60(5):503–520, 2004.

Stephen Robertson, Hugo Zaragoza, et al. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389, 2009.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. *arXiv preprint arXiv:1508.07909*, 2015.

Chenyan Xiong, Zhuyun Dai, Jamie Callan, Zhiyuan Liu, and Russell Power. End-to-end neural ad-hoc ranking with kernel pooling. In *Proceedings of the 40th International ACM SIGIR conference on research and development in information retrieval*, pages 55–64. ACM, 2017.

Hamed Zamani, Bhaskar Mitra, Xia Song, Nick Craswell, and Saurabh Tiwary. Neural ranking models with multiple document fields. In *Proc. WSDM*, 2018.