Understanding and Improving Neural Ranking Models from a Term Dependence View

Yixing Fan(✉), Jiafeng Guo, Yanyan Lan, and Xueqi Cheng

CAS Key Lab of Network Data Science and Technology,
Institute of Computing Technology, Chinese Academy of Sciences,
Beijing 100190, China
{fanyixing,guojiafeng,lanyanyan,cxq}@ict.ac.cn

Abstract. Recently, neural information retrieval (NeuIR) has attracted a lot of interests, where a variety of neural models have been proposed for the core ranking problem. Beyond the continuous refresh of the state-of-the-art neural ranking performance, the community calls for more analysis and understanding of the emerging neural ranking models. In this paper, we attempt to analyze these new models from a traditional view, namely term dependence. Without loss of generality, most existing neural ranking models could be categorized into three categories with respect to their underlying assumption on query term dependence, i.e., independent models, dependent models, and hybrid models. We conduct rigorous empirical experiments over several representative models from these three categories on a benchmark dataset and a large click-through dataset. Interestingly, we find that no single type of model can achieve a consistent win over others on different search queries. An oracle model which can select the right model for each query can obtain significant performance improvement. Based on the analysis we introduce an adaptive strategy for neural ranking models. We hypothesize that the term dependence in a query could be measured through the divergence between its independent and dependent representations. We thus propose a dependence gate based on such divergence representation to softly select neural ranking models for each query accordingly. Experimental results verify the effectiveness of the adaptive strategy.

Keywords: Understanding · Term dependence · Query adaptation

1 Introduction

Recently, deep neural networks have led to exciting breakthroughs in speech recognition, computer vision, and natural language processing (NLP) tasks. This also inspires researchers to apply neural models for the core ranking problem in the information retrieval (IR) community. During the past few years, a large number of neural ranking models have been proposed, leading to a hot topic

F. L. Wang et al. (Eds.): AIRS 2019, LNCS 12004, pp. 118–130, 2020.
https://doi.org/10.1007/978-3-030-42835-8_11

named NeuIR. However, beyond continuous refresh of the state-of-the-art neural ranking performance, the community calls for more analysis and understanding of the emerging neural ranking models.

There have been a few studies making progress in understanding the architecture of neural ranking models. For example, in [5], the authors categorized existing neural ranking models into two types according to their model architecture, namely representation-focused models and interaction-focused models. Mitra et al. [15] also provided similar idea, but named the two categories as lexical matching models and semantic matching models. They show in general interaction-focused models work better than representation-focused models since ranking is more directly about interaction between the query and the document. In [3], the authors studied different granularity of IR tasks to analyze what information is important or extraneous at each level of a neural ranking model.

In this paper, we try to analyze neural ranking models from a different dimension. Unlike previous works [5,15] which categorize neural ranking models mainly based on model architecture, we take a traditional IR view, namely term dependence view, to look at these existing neural ranking models. Term dependence has been a long-studied problem in IR. It has been widely accepted that it is of great importance to model the term dependence in an effective retrieval model [2,13,24]. In [13], the authors have introduced three term independence assumptions, namely full independence, sequential dependence, and full dependence, under the framework of Markov random field.

When we look at existing neural ranking models from the term dependence view, we find that these models can be categorized into three groups, namely independent model, dependent model, and hybrid model. Although the existing neural ranking models do not mention term dependence in their model design, they actually take one of the three underlying assumptions on term dependence. We then conduct rigorous empirical comparisons to analyze the three categories of models based on both a benchmark LETOR4.0 data and a large scale click-through data collected from a commercial search engine. We find that there is no clear winner between the three types of models. Even the hybrid model does not show consistent advantages as one may expect. Moreover, beyond the average performance, we also look at the detailed performance on each query. We find that each category of models have their own advantages and perform better on a subset of queries. If there is an oracle that can select a right model for each query, we can significantly improve the ranking performance. This indicates that there is a large room for the design or optimization of neural ranking models.

Based on the above observations, we introduce an adaptive strategy for neural ranking models, which attempts to select neural models with different dependence assumption for each query accordingly. Specifically, we hypothesize that the term dependence in a query could be measured through certain divergence between its independent and dependent representations. We propose a term dependence gate based on such divergence, and use it to softly select between an independent and a dependent neural ranking model for each query. We evaluate the effectiveness of the proposed adaptive strategy using the same two datasets

mentioned above. The experimental results demonstrate that by adapting to each query with respect to term dependence, one can obtain significant performance improvement.

2 Related Work

In this section, we briefly review the studies relevant to our work, including understanding on neural ranking models and term dependence in retrieval models.

2.1 Understanding on Neural Ranking Models

There have been a few efforts to understand the neural ranking models. For example, Guo et al. [5,7] has analyzed the architecture of the existing neural ranking models, and categorized these models into different groups. Cohen et al. [3] proposed a probe based methodology to evaluate what information is important or extraneous at each level of a network. Nie et al. [17] proposed to compare the representation-focused models and interaction-focused models under the same condition. They built a similar convolution networks to learn either representations and interaction patterns between query and document. Though several works have made their efforts in understanding the neural ranking models, to the best of our knowledge, there are no works trying to understand the neural ranking models from the term dependence view.

2.2 Term Dependence in Retrieval Models

Different dependence assumptions between query terms have been made in designing retrieval models. Note here term dependence, broadly speaking, is also known as *term co-occurrence*, *adjacency*, *lexical affinities*, and *proximity* in IR. Without loss of generality, existing models can be divided into three classes, namely independent model, dependent model, and hybrid model, according to the degree of the underlying dependence assumptions.

Firstly, the independent models assume each query term independent from others. In this way, to compute the relevance of a document, one can first estimate the matching between the document and each query term separately, and aggregate these matching signals to produce the final relevance score. A large number of models have been designed under this branch [22,25,28]. Although many independent models (e.g., BM25 and QL) are simple and effective on different queries, they are often considered insufficient by ignoring the term dependencies (such as *information* and *retrieval*) which may help filter out irrelevant document efficiently [13]. Obviously, it is insufficient to treat each term independently as the term dependence exists in queries everywhere. Secondly, the dependent models assume query terms be dependent on each other in some way [13]. In this way, the relevance score can no longer be decomposed to each query term, but rather be computed with respect to dependent units, such as

phrases, n-grams or even the whole query [1,9,23,24]. For example, The bi-term language model [24] attempts to capture the term dependence between term pairs under the language model framework. Although the dependence assumption seems more powerful than the independence assumption, the performance of the dependent models is not consistently the best so far as we know. The possible reason is that there is very little hope of accurately modeling general term dependencies due to data sparsity, if at all [13]. Lastly, the hybrid models propose to combine both assumptions to improve the retrieval performance. There have been a number of retrieval models developed in this manner [12,13,20]. For example, in [13], Metzler et al. constructs a Markov random field on query terms which models multiple query term dependencies (i.e., single terms, ordered phases, and unordered phrases) simultaneously. Although hybrid models take into account multiple dependence assumptions, they actually pose a strong underlying assumption that some fixed combination of independence and dependence assumptions could fit all the queries.

3 Dependence View of Neural Ranking Models

In this section, we first introduce the dependence view of the neural ranking models. Then, we conduct experiments to analyze existing models with different dependence assumptions.

3.1 Dependence Categorization

There have been a few taxonomies proposed for existing neural ranking models. For example, in [5], the neural ranking models are categorized into representation-focused and interaction-focused model based on their architectures. Different from the architecture view, we look at existing neural ranking models from the dependence view, which have been mainly investigated over traditional retrieval methods [2,13]. Although existing neural ranking models do not mention the term dependence in their model design, they actually take a specific assumption on term dependence. Without loss of generality, existing neural ranking models can be categorized into three categories, namely independent models, dependent models, and hybrid models. The independent model, as its name suggested, assumes independence between terms. In this way, the relevance score could be decomposed with respect to each query term. Representative models include:

- **DRMM:** The DRMM [5] treats both query and document as bag of word embeddings. Each query term interacts with the document to produce the term level matching score.
- **K-NRM:** The K-NRM [27] is a neural ranking model built upon DRMM, which uses a kernel pooling layer to replace the matching histogram layer in DRMM.

The dependent model, assumes the terms are in some way dependent on each other. More specifically, according to the range of term dependence, there are two types of dependence, namely partially dependent and fully dependent. The partial dependent model assumes the terms are dependent on each other within a local contextual window. Representative models include:

- **ARCII:** The ARCII [8] utilizes a one-dimensional convolution neural network to enhance the term representation by a local context, where the window size determines the range of dependent scope.
- **MatchPyramid:** The MatchPyramid [18] constructs a matching matrix based on term-term interactions. Then, a two dimensional convolution neural network is applied on the matching matrix to capture the proximity between terms in pre-defined size windows.

The fully dependent model assumes all terms are dependent with each other. Representative models include:

- **ARCI:** The ARCI [8] firstly learns the global representation for both query and document. Then, the final score is obtained based on the interaction between these two representations.
- **DSSM:** The DSSM [9] also learned a global representation for each text, but employs a fully connected neural network on the tri-letters instead.

Other models like CDSSM [23] and MVLSTM [26] also belong to this category.

The hybrid model considers both assumptions simultaneously and combines models from different categories. Representative models include:

- **PACRR:** The PACRR [10] captures both the independent (i.e., unigram) term matching and the dependent (i.e., n-gram) term matching by convolution neural networks.
- **Duet:** The Duet [16] combines a local model with a distributed model for query-document matching. The local model captures the term level interaction independently and the distributed model learns global representation for both query and document in a fully dependent way.

Some other models such as Conv-KNRM [4], MatchTensor [11], and DeepRank [19] also fall into this category.

3.2 Experimental Setting

To better understand the characteristics of models with different dependence assumptions, we conduct empirical analysis over representative models on benchmark datasets.

Data Sets. To compare the results of different dependent models, we conduct experiments on LETOR4.0 dataset [21] and a large scale click-through dataset. We choose these two datasets since they contain sufficiently large collections of

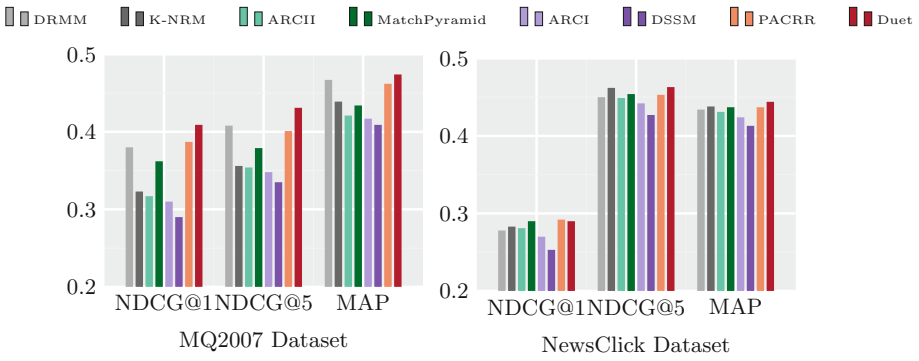

Fig. 1. Performance comparison of different dependent models on two datasets.

queries, which are desirable for training and comparing many data-hungry (i.e., neural) retrieval models. Specifically, in LETOR4.0, we leverage the MQ2007 dataset as the testbed since it contains much more queries than MQ2008. The click-through data, namely NewsClick, is collected from a commercial news search engine, where clicked documents are viewed to be relevant, and the others are viewed as irrelevant. We apply some typical data pre-processing techniques, such as word segmentation, stopping words and low frequency words (less than 100) removing. After these preprocessing, the final NewsClick dataset contains 223,783 queries and 6,292,911 documents.

Evaluation Methodology. For MQ2007, We follow the data partition in Letor4.0 [21], and 5-fold cross-validation is conducted to minimize overfitting as in [5]. Specifically, the parameters for each model are tuned on 4-of-5 folds. The last fold in each case is used for evaluation. For NewsClick, we partitioned the dataset into training/validation/testing sets according to the proportion 8:1:1. Here, we adopt normalized discounted cumulative gain (NDCG) and mean average precision (MAP) as the evaluation metrics.

Model Details. Here, we choose two representative models from different categories to conduct the experiments. Specifically, we choose DRMM and K-NRM as the independent model. For dependent model, we selected two representative models for both the partial dependent model and the fully dependent model, i.e., MatchPyramid and ARCII as the partial dependent model, and ARCI and DSSM as the fully dependent model. For the hybrid model, we choose the PACRR and Duet. The implementations of these models are based on the open-source toolkit MatchZoo [6]. To train these models for the MQ2007 dataset, we have utilized the pre-trained term vectors on the Wikipedia corpus[1] using the CBOW model [14]. All other trainable parameters are randomly initialized by uniform distribution within $[-0.2, 0.2]$.

[1] http://en.wikipedia.org/wiki/Wikipediadatabase.

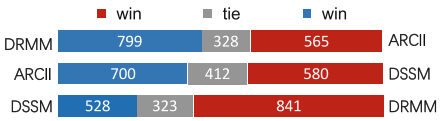

Fig. 2. The pairwise comparison of different models on MQ2007 dataset. DRMM is a independent model, ARCII is a partial dependent model, DSSM is a fully dependent model.

Model Name	NDCG@1	NDCG@5	MAP
DRMM	0.380	0.408	0.467
ARCII	0.317	0.354	0.449
DSSM	0.290	0.335	0.409
ORACLE	0.497	0.528	0.542

Fig. 3. Performance comparison of different retrieval models on MQ2007 dataset.

3.3 Empirical Analysis

The overall results are depicted in the Fig. 1. We have the following observations:

1. For the independent model, we found that the performance winner is not consistent between DRMM and K-NRM on different datasets. K-NRM can outperform DRMM on the larger dataset (i.e., NewsClick) when the word embeddings can be learned in an end-to-end way, but may not work well as DRMM when the dataset is relatively small (i.e., MQ2007).
2. For the dependent model, the partially dependent models are always better than the fully dependent models in terms of all evaluation metrics. This might be due to the fact that modeling full dependence is much more complicated than modeling partial dependence, since the sparsity problem becomes much more severe in the full dependence [13].
3. For the hybrid models, we found that Duet outperforms the PACRR on MQ2007 dataset in terms of all the three metrics. However, on NewsClick dataset, the PACRR performs better than Duet in terms of NDCG@1.
4. When comparing the four groups of models, the hybrid models (i.e., PACRR and Duet) in general can perform better than the independent models (i.e., DRMM and K-NRM) and the dependent models (i.e., MatchPyramid and ARCII). However, there are still some exceptions. For example, DRMM outperforms PACRR on MQ2007 in terms of NDCG@5 and MAP.

From the above results, we find that models with different dependence assumptions have their own advantages. There is no single model, with a fixed assumption, that can achieve the best performance over all the datasets. Here we further conduct some detailed comparisons between pairs of neural ranking models with different dependence assumptions on MQ2007. For each pair of models, we report the number of queries over which one model performs better (i.e., "win") or the same (i.e., "tie") as compared with the other. From the results in Fig. 2 we find that each model have their own advantages on a specific group of queries. For example, when compare DRMM with ARCII, there are about 799 queries which DRMM performs better than ARCII. However, there are also 565 queries where ARCII gets higher performance. Similarly, the conclusion can be drawn from the other two pairs.

It is not surprising to see that models with a specific dependence assumption can fit well on queries which share the same dependence assumption. In consequence, when a model only takes a specific assumption on the term dependence, it may inevitably fail on queries that do not fit that assumption. Therefore, an intuitive way is to select the right dependent model for each query. As shown in Fig. 3, if we have an oracle that can always select the best model among the three (e.g., in terms of MAP) for each query adaptively, the retrieval performance would be significantly boosted. Based on the above analysis, we argue that rather than using a pre-determined dependence assumption, a better ranking strategy is to adapt to each query with the right dependence assumption.

4 Dependence-Based Query Adaptation Strategy

In this section, we introduce an adaptive strategy for neural ranking models with respect to term dependence. The key idea is as follows. Since retrieval models under different dependence assumptions may fit different queries, we attempt to learn to measure the degree of term dependence in a query, and use this measure to select retrieval models with the right dependence assumption for each query.

In an abstract level, we consider two types of neural ranking models, i.e., independent models and dependent models, as the basic components. Then, a term dependence gate is employed to softly select between them adaptively. For the independent model, we choose a variant of the DRMM as the implementation. Specifically, we replace the matching histogram with a sorted top-k pooling layer [10], where the strongest signals are always placed at the top positions to keep the strength preserving property. In this way, the varied DRMM can be learned in an end-to-end way. For the dependent model, we choose two existing neural models, i.e., MatchPyramid [18] and ARCI [8] as the partially dependent model and fully dependent model respectively. In the following, we will describe the term dependence gate, which is the key component in our adaptive strategy.

4.1 Term Dependence Gate

The term dependence gate attempts to measure the dependence degree between query terms, and use this measure to softly select the above sub-models with different dependence assumptions adaptively. The key idea of the term dependence measure is as follows. If we assume no dependence between query terms, the meaning of a query is a union of its terms. In other words, we may obtain a query representation by some simple aggregation of its term representations. We name this representation of a query as its *fully-independent representation*. If we assume dependence between query terms, the meaning of a query then becomes a union of its dependent units. In an extreme case, i.e., the full dependence assumption, all query terms are dependent to each other in some way. In this case, the meaning of a query can no longer be decomposed into smaller units. We may obtain the query representation from its term representations through some complicated semantic interactions. We name this representation of a query

as its *fully-dependent representation*. If we find that the fully-dependent representation of a query is very close to its fully-independent representation, it indicates that there might be very weak or even no dependence between query terms. On the contrary, if we find that the fully-dependent representation of a query is significantly different from its fully-independent representation, it indicates that there might exist strong dependence between query terms.

Based on the above ideas, we design the following term dependence gating network. Specifically, we firstly obtain the fully-independent representation $\mathbf{q}_{ind}$ using a simple sum over its term embeddings. To obtain its fully-dependent representation $\mathbf{q}_{dep}$, we employ a CNN over its term representations to capture the complicated semantic interactions. Given these two query representations, we take a simple but effective way by directly taking the difference $\mathbf{q}_{dep} - \mathbf{q}_{ind}$ as the input, and feed it into a feed forward neural network to form the gate. In this way, the final gating function is as follows:

$$g(\mathbf{Q}) = \sigma(\mathbf{W}_g[\mathbf{q}_{dep} - \mathbf{q}_{ind}]^T + \mathbf{b}_g), \tag{1}$$

where $\mathbf{W}_g$ and $\mathbf{b}_g$ are parameters to be learned, and σ is the sigmoid activation function to keep the value of gate among $[0, 1]$.

Finally, we use this term dependence gating network to softly select the two sub-models and obtain the relevance score by

$$f(\mathbf{Q}, \mathbf{D}) = g(\mathbf{Q}) \cdot f_i(\mathbf{Q}, \mathbf{D}) + (1 - g(\mathbf{Q})) \cdot f_d(\mathbf{Q}, \mathbf{D}). \tag{2}$$

where $f_i(\mathbf{Q}, \mathbf{D})$ and $f_d(\mathbf{Q}, \mathbf{D})$ denote the output score of the independent model and the dependent model, respectively.

4.2 Model Training

The introduced adaptive model can be learned in an end-to-end way. We utilize the pairwise hinge loss to train our model:

$$\mathcal{L}(\mathbf{Q}, \mathbf{D}^+, \mathbf{D}^-; \theta) = \max(0, 1 - f(\mathbf{Q}, \mathbf{D}^+) + f(\mathbf{Q}, \mathbf{D}^-))$$

where $f(\mathbf{Q}, \mathbf{D})$ denotes the relevance score and D^+ ranks higher than D^-. θ includes all the parameters to be learned.

5 Experiment

In this section, we conduct experiments to verify the effectiveness of the adaptive model based on the same MQ2007 and NewsClick datasets, which have been introduced in the previous section.

5.1 Experimental Settings

We refer to our proposed model as **ADNR** (i.e., Adaptive Neural Ranking). Since the dependent sub-module could be a partially dependent model or a fully dependent model, we refer to these two variants as $ADNR_{PD}$ and $ADNR_{FD}$, respectively. For the network configurations (e.g. number of layers and hidden nodes), we tuned the hyper-parameters via the validation set. Specifically, the embedding size is set to 50. In the independent model, the k in top-k pooling layer is set to 100 and 20 On MQ2007 and NewsClick as their document length differs significantly, and the multi-layer perceptron is set to 3 layers with the hidden size set to 10. In the dependent model, we have 64 kernels with size 3×3 in the convolution layer, set the max pooling size to 3×5, and use a 2-layer perceptron for output. We perform significant tests using the paired t-test. Differences are considered statistically significant when the p-value is lower than 0.01. All other trainable parameters are randomly initialized by uniform distribution within $[-0.2, 0.2]$.

In addition to the neural ranking models, we also include several traditional retrieval models as baselines: (1) BM25 [22] is a classic and highly effective independent model. (2) PDFR [20] is a partially dependent model, which assumes adjacent query terms are dependent. (3) SD [13] is a fully dependent model which utilize the Markov random field to model the sequential dependence. (4) WSD [2] is a hybrid model which combines a fully independent model, a sequentially dependent model, and a fully dependent model with handcrafted features.

5.2 Overall Comparison

In this section we compare the ADNR models against all the baselines on the two datasets. A summary of the main results is displayed in Table 1.

Firstly, for the independent models, we can see that DRMM is a strong baseline which performs better than traditional ranking model (i.e., BM25). K-NRM can obtain better performance when in larger dataset (i.e., NewsClick). Secondly, for the dependent models, we find that the traditional retrieval model, i.e., PDFR and SD, can outperform the neural dependent models on MQ2007 dataset, but become worst on NewsClick. It indicates that when there are sufficient data, the neural dependent models could better capture the term dependence patterns and achieve better performance than traditional dependent models. Thirdly, for the hybrid models, we can see that WSD performs significantly better than other traditional models such as BM25, PDFR, and SDM by taking into account both uni-gram matching and dependent term matching. However, it is still less effective than PACRR and Duet, which can capture more complex term dependence patterns through deep neural networks. Overall, we find Duet the best performing model among all the baseline methods by linearly combining a dependence sub-model and an independence sub-model. Finally, we observe that the two variants of ADNR can achieve better performances than all baseline methods. For example, on NewsClick, the relative improvement of ADNR_{IP} over the best performing baseline (i.e., Duet) is about 8.1% in terms of MAP. For the two

Table 1. Comparison of different retrieval models over the MQ2007 and NewsClick datasets. Significant improvement or degradation with respect to ADNR$_{IP}$ is indicated (+/−) (p-value ≤ 0.01).

Model name	MQ2007			NewsClick		
	NDCG@1	NDCG@5	MAP	NDCG@1	NDCG@5	MAP
BM25	0.358^{-}	0.384^{-}	0.450^{-}	0.207^{-}	0.385^{-}	0.378^{-}
DRMM	0.380^{-}	0.408^{-}	0.467^{-}	0.278^{-}	0.450^{-}	0.433^{-}
K-NRM	0.323^{-}	0.356^{-}	0.439^{-}	0.283^{-}	0.461^{-}	0.438^{-}
PDFR	0.345^{-}	0.371^{-}	0.442^{-}	0.223^{-}	0.415^{-}	0.393^{-}
MatchPyramid	0.362^{-}	0.379^{-}	0.434^{-}	0.290^{-}	0.454^{-}	0.437^{-}
ARCII	0.317^{-}	0.354^{-}	0.421^{-}	0.281^{-}	0.449^{-}	0.431^{-}
SD	0.383^{-}	0.395^{-}	0.455^{-}	0.248^{-}	0.421^{-}	0.408^{-}
ARCI	0.310^{-}	0.348^{-}	0.417^{-}	0.270^{-}	0.442^{-}	0.422^{-}
DSSM	0.290^{-}	0.335^{-}	0.409^{-}	0.253^{-}	0.427^{-}	0.413^{-}
WSD	0.385^{-}	0.399^{-}	0.457^{-}	0.249^{-}	0.423^{-}	0.410^{-}
PACRR	0.387^{-}	0.401^{-}	0.462^{-}	0.292^{-}	0.453^{-}	0.437^{-}
Duet	0.409	0.431	0.474^{-}	0.290^{-}	0.463^{-}	0.444^{-}
ADNR$_{IF}$	0.408	0.431	0.480	0.330^{-}	0.498	0.474
ADNR$_{IP}$	**0.413**	**0.439**	**0.487**	**0.337**	**0.500**	**0.480**

variants of ADNR, the ADNR$_{IP}$ could consistently outperform the ADNR$_{IF}$, this may due to the fact that the partial dependent model is more effective than the fully dependent model. Meanwhile, it is noteworthy that the ADNR is built upon an independent model (i.e., DRMM) and a dependent model (e.g., MatchPyramid and ARCI) with a term dependence gating network. When we compare ADNR with its sub-models, we can see that the performance can be significantly improved through adaptive combination, e.g., on NewsClick, the relative improvement of ADNR$_{IP}$ over DRMM and MatchPyramid is about 10.9% and 9.8% in terms of MAP, respectively. All these results demonstrate the effectiveness of the adaptive strategy.

6 Conclusions

In this paper, we try to understand the neural ranking models from the term dependence view. In this way, The neural ranking models are categorized into three classes according to the underlying assumption on the term dependence. Moreover, we conducted rigorous empirical comparisons over three categories of models, and find that on one category of models can achieve best performance for all queries. We proposed a novel term dependence gate which learns to measure the term dependence degree in the query. Experimental results on a benchmark dataset and a large click-through dataset demonstrate the effectiveness of the

adaptive strategy. For future work, we will try to employ natural language processing methods, e.g., dependency grammar analysis and syntactic analysis, to measure the term dependence.

Acknowledgements. This work was funded by the National Natural Science Foundation of China (NSFC) under Grants No. 61902381, 61425016, 61722211, 61773362, and 61872338, the Youth Innovation Promotion Association CAS under Grants No. 20144310, and 2016102, the National Key R&D Program of China under Grants No. 2016QY02D0405, and the Foundation and Frontier Research Key Program of Chongqing Science and Technology Commission (No. cstc2017jcyjBX0059).

References

1. Bendersky, M., Kurland, O.: Utilizing passage-based language models for document retrieval. In: Macdonald, C., Ounis, I., Plachouras, V., Ruthven, I., White, R.W. (eds.) ECIR 2008. LNCS, vol. 4956, pp. 162–174. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78646-7_17
2. Bendersky, M., Metzler, D., Croft, W.B.: Learning concept importance using a weighted dependence model. In: WSDM, pp. 31–40. ACM (2010)
3. Cohen, D., O'Connor, B., Croft, W.B.: Understanding the representational power of neural retrieval models using NLP tasks. In: SIGIR, pp. 67–74. ACM (2018)
4. Dai, Z., Xiong, C., Callan, J., Liu, Z.: Convolutional neural networks for soft-matching n-grams in ad-hoc search. In: WSDM, pp. 126–134. ACM (2018)
5. Guo, J., Fan, Y., Ai, Q., Croft, W.B.: A deep relevance matching model for ad-hoc retrieval. In: CIKM, pp. 55–64. ACM (2016)
6. Guo, J., Fan, Y., Ji, X., Cheng, X.: MatchZoo: a learning, practicing, and developing system for neural text matching. In: Proceedings of the 42Nd International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR 2019, pp. 1297–1300. ACM, New York, NY, USA (2019)
7. Guo, J., et al.: A deep look into neural ranking models for information retrieval. arXiv preprint arXiv:1903.06902 (2019)
8. Hu, B., Lu, Z., Li, H., Chen, Q.: Convolutional neural network architectures for matching natural language sentences. In: NIPS, pp. 2042–2050 (2014)
9. Huang, P.-S., He, X., Gao, J., Deng, L., Acero, A., Heck, L.: Learning deep structured semantic models for web search using clickthrough data. In: CIKM, pp. 2333–2338. ACM (2013)
10. Hui, K., Yates, A., Berberich, K., de Melo, G.: A position-aware deep model for relevance matching in information retrieval. CoRR (2017)
11. Jaech, A., Kamisetty, H., Ringger, E., Clarke, C.: Match-tensor: a deep relevance model for search. arXiv preprint arXiv:1701.07795 (2017)
12. Lioma, C., Simonsen, J.G., Larsen, B., Hansen, N.D.: Non-compositional term dependence for information retrieval. In: SIGIR, pp. 595–604. ACM (2015)
13. Metzler, D., Croft, W.B.: A Markov random field model for term dependencies. In: SIGIR, pp. 472–479. ACM (2005)
14. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS, pp. 3111–3119 (2013)
15. Mitra, B., Craswell, N.: Neural models for information retrieval. arXiv preprint arXiv:1705.01509 (2017)

16. Mitra, B., Diaz, F., Craswell, N.: Learning to match using local and distributed representations of text for web search. In: WWW, pp. 1291–1299. International World Wide Web Conferences Steering Committee (2017)
17. Nie, Y., Li, Y., Nie, J.-Y.: Empirical study of multi-level convolution models for IR based on representations and interactions. In: SIGIR, pp. 59–66. ACM (2018)
18. Pang, L., Lan, Y., Guo, J., Xu, J., Wan, S., Cheng, X.: Text matching as image recognition. In: AAAI, pp. 2793–2799 (2016)
19. Pang, L., Lan, Y., Guo, J., Xu, J., Xu, J., Cheng, X.: DeepRank: a new deep architecture for relevance ranking in information retrieval. In: CIKM, pp. 257–266. ACM (2017)
20. Peng, J., Macdonald, C., He, B., Plachouras, V., Ounis, I.: Incorporating term dependency in the DFR framework. In: SIGIR, pp. 843–844. ACM (2007)
21. Qin, T., Liu, T.-Y., Xu, J., Li, H.: LETOR: a benchmark collection for research on learning to rank for information retrieval. Inf. Retr. **13**(4), 346–374 (2010)
22. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In: Croft, B.W., van Rijsbergen, C.J. (eds.) SIGIR, pp. 232–241. Springer, London (1994). https://doi.org/10.1007/978-1-4471-2099-5_24
23. Shen, Y., He, X., Gao, J., Deng, L., Mesnil, G.: A latent semantic model with convolutional-pooling structure for information retrieval. In: CIKM, pp. 101–110. ACM (2014)
24. Srikanth, M., Srihari, R.: Biterm language models for document retrieval. In: SIGIR, pp. 425–426. ACM (2002)
25. Turtle, H., Croft, W.B.: Evaluation of an inference network-based retrieval model. TOIS **9**(3), 187–222 (1991)
26. Wan, S., Lan, Y., Guo, J., Xu, J., Pang, L., Cheng, X.: A deep architecture for semantic matching with multiple positional sentence representations. In: AAAI, vol. 16, pp. 2835–2841 (2016)
27. Xiong, C., Dai, Z., Callan, J., Liu, Z., Power, R.: End-to-end neural ad-hoc ranking with kernel pooling. In: SIGIR, pp. 55–64. ACM (2017)
28. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to ad hoc information retrieval. In: ACM SIGIR Forum, vol. 51, pp. 268–276. ACM (2017)