Neural Retrieval with Partially Shared Embedding Spaces

Bo Li, Le Jia
School of Computer Science, Central China Normal University
Wuhan, China
libo@mail.ccnu.edu.cn

ABSTRACT

One category of neural information retrieval models tries to learn text representation in a common embedding space for both queries and documents. However, a single embedding space is not always sufficient, since queries and documents are different in terms of length, number of topics covered, etc. We argue that queries and documents should be mapped into different but overlapping embedding spaces, which is named Partially Shared Embedding Space (PSES) model in this paper. PSES consists of two embedding spaces respectively for queries and documents, and a shared embedding space capturing common features of two sources. Those three embeddings are learned by jointly obeying three constraints: a feature separation constraint, a pairwise matching constraint, and a reconstruction constraint. Experiments on standard TREC collections indicate that PSES leads to significant better performance of retrieval over traditional IR models and several neural IR models with only one embedding space.

CCS CONCEPTS

• **Information systems → Retrieval models and ranking**;

KEYWORDS

Neural retrieval; Shared embedding space; Adversarial learning

ACM Reference Format:
Bo Li, Le Jia. 2018. Neural Retrieval with Partially Shared Embedding Spaces. In *The 27th ACM International Conference on Information and Knowledge Management (CIKM '18), October 22–26, 2018, Torino, Italy.* ACM, New York, NY, USA, 4 pages. https://doi.org/10.1145/3269206.3269306

1 INTRODUCTION

Neural networks have shown promising performance in computer vision and natural language processing tasks. Accordingly, the IR community has developed in very recent years a significant number of neural IR models (e.g., [4, 7, 19]). One category of such models relies on the idea of *semantic matching* which seeks a low-dimensional and semantic-rich subspace where queries and documents can be mapped and compared.

CIKM '18, October 22–26, 2018, Torino, Italy

ACM ISBN 978-1-4503-6014-2/18/10…$15.00
https://doi.org/10.1145/3269206.3269306

Depending on if the embeddings are learned with query-document relevance information, exiting studies can be divided as unsupervised and supervised approaches. Unsupervised approaches use general word embeddings pre-trained with tools such as *word2vec* to enhance traditional IR models (e.g., [5, 17]), which tries to capture term proximity rather than relevance that is essential for retrieval. Supervised approaches rely on query-document relevance signals to learn task-specific representations optimized for IR (e.g., [4, 7]), which reflects a more significant shift toward pursuing an *end-to-end* framework for IR.

Supervised learning approaches account better for retrieval requirements and have been favored by most recent studies [4, 18]. However, one can note that existing studies prefer to map both queries and documents into a single embedding space. Intuitively, queries and documents are different in terms of length, number of topics covered, etc. We thus argue in this paper that queries and documents should be treated differently in IR task. Inspired by the multi-task learning framework in [2, 9], we plan to map queries and documents into different but overlapping embedding spaces, which is named Partially Shared Embedding Space (PSES). The shared embedding space aims to capture common features of two sources while query/document specific embedding space aims to capture those features only relevant to a specific source. PSES is advantageous since it explicitly models the relationship between queries and documents in the text representation process for retrieval via embedding space splitting. To the best of our knowledge, such an idea has never been investigated in IR.

To this end, we define three constraints to direct the three embedding networks to encode their respective features, which are (1) a feature separation constraint that encourages the three embedding networks to encode different aspects of the inputs and to be separate from each other. (2) a pairwise matching constraint that forces embedding spaces to account for task-specific characteristics of IR. (3) a reconstruction constraint that guarantees no information is lost during embedding space splitting.

2 RELATED WORK

Neural networks have recently gained significant popularity in information retrieval. Apart from learning to rank approaches that train their models over a set of hand-crafted features [10], neural IR models can learn the hidden structures and features from the raw text at different levels of abstraction. Text representation can be learned in an unsupervised manner with unlabeled corpora or in a supervised fashion with query-document relevance information.

Unsupervised approaches rely on unlabeled data to learn general text embeddings that can be used to extend traditional IR models. Ganguly et al. [5] propose a generalized language model with query-likelihood language modeling for integrating word embeddings learned by *word2vec*, a popular tool also used in neural IR

studies such as [17, 19]. There are also works developing their own embedding methodologies for IR. For instance, Clinchant and Perronnin [3] use latent semantic indexing to induce text embeddings.

Supervised approaches use query-document relevance information to learn a representation that is optimized for IR. Huang et al. [7] develop DSSM, which is a feed-forward neural network with a word hashing phrase as the first layer to predict the click probability given a query and a document from click-through data. DSSM is extended in [14, 15] by incorporating CNN and max-pooling layers to extract the most salient local features. Since click-through data are not always available out of industrial labs due to user privacy reasons, the most recent studies use pseudo-labels as weak supervision. Dehghani et al. [4] employ BM25 to label relevant documents for AOL queries, which is then used as supervision for joint embedding and ranking model training. Zamani and Croft [18] obtain their pseudo-labels in a similar way as [4] and train a *word2vec*-style embedding network based on such labels. In this paper, we concentrate more on semantic matching models that can be combined with interaction-focused models to achieve further improvement [11].

Supervised approaches are superior since they explicitly capture relevance so as to account better for retrieval requirements. However, we note that existing works mostly use a single latent space to encode queries and documents. It is far from optimal in the context of IR, because queries and documents are different in terms of length, number of topics covered, etc. We thus argue that queries and documents should be treated differently when obtaining their latent representations. We are inspired by recent advances in multi-task learning [2, 9] and propose to map queries and documents into different but overlapping embedding spaces.

3 PARTIALLY SHARED EMBEDDING SPACE MODEL

We will detail in this section the partially shared embedding space (PSES) model.

General framework

As we have discussed in section 1, PSES consists of three embedding networks: one query-specific network NN_q encoding the input query x_q into its query-specific embedding form, one document-specific network NN_d that embeds the document x_d to the document-specific latent form, and one shared network NN_s that maps both x_q and x_d into the shared embedding space. The framework is illustrated in figure 1. In addition to the three encoding networks, we have two decoding networks NN_{rq} and NN_{rd} respectively for queries and documents, and an adversarial network NN_{adv} that tries to detect the source (query or document) of the shared embedding vector z_s to facilitate adversarial training. The details of these networks will be given below. On top of the framework, we consider three constraints in order to learn the partially shared embedding vectors optimized for the retrieval task.

Feature separation constraint

A crucial requirement behind the PSES model is that different embedding spaces do not overlap with each other. This goal can be achieved by simultaneously taking two actions: (1) purify the shared embedding space by eliminating query/document specific features, and (2) remove shared features from the query/document

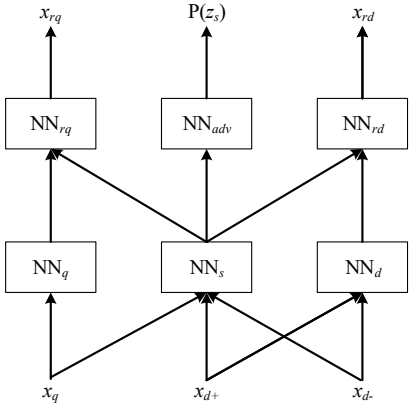

Figure 1: The framework of partially shared embedding space (PSES) model.

embedding spaces. The same idea has been tested in multi-task learning literature [9], which has never been investigated in IR.

In order to purify the shared embedding space, we follow the convention in GAN [6] and develop an adversarial component on top of the shared embedding space so that a discriminator model tries its best to detect whether a shared representation z_s is produced from x_q or x_d. Different from previous studies [2, 9], we propose to use WGAN [1] to overcome the training difficulty problem of GAN [6] due to the minimization of Jensen-Shannon (JS) divergence. WGAN uses the Earth Mover's Distance instead of JS to measure the distance between the real data distribution P_r and the generative distribution P_g. With approximation and Kantorovich-Rubinstein duality [16], the WGAN value function can be written as:

$$\min_G \max_{D \in \mathbb{D}} [E_{x \sim P_r} D(x) - E_{x \sim P_g} D(x)] \tag{1}$$

where G and D respectively denote the generator and discriminator, $\mathbb{D}$ is the set of 1-Lipschitz functions. The derivation details are omitted here for space reasons. To enforce the Lipschitz constraint for D, weight clipping [1] has been used in the iterative optimization of the minmax game. In our model, the discriminator D is in the form of an adversarial classifier NN_{adv} implemented as a neural network with softmax activation after the last layer. The output of NN_{adv} then corresponds to a probability distribution vector over query and document. Let us denote the ground truth label of the current source (query or document) as a vector y. In this case, we can adjust equation 1 to our settings and obtain the adversarial loss L_{adv} on a query set Q and a document set D, which is:

$$L_{adv} = \min_G \max_D \sum_{x \in Q,D} (-1)^{\mathbb{I}(x)} \cdot NN_{adv}(z_s) \circ y \tag{2}$$

where $\circ$ is the inner product operator, and $\mathbb{I}(x)$ is the indicator function to determine if x belongs to the document set.

In addition, the query/document specific embedding space may contain shared features. We make use here an orthogonality constraint that was proposed in [2]. For a query set Q, let us build a

matrix $\mathbb{Z}_{sQ}$ (resp. $\mathbb{Z}_Q$) by stacking the shared representation (resp. query-specific representation) of each query q as a row. For a document set D, we can build matrices $\mathbb{Z}_{sD}$ and $\mathbb{Z}_D$ in the same way. The orthogonality loss L_{ort} on Q and D can be defined as the sum of two Frobenius norms, which is:

$$L_{ort} = \|\mathbb{Z}_{sQ}^T \cdot \mathbb{Z}_Q\|_F^2 + \|\mathbb{Z}_{sD}^T \cdot \mathbb{Z}_D\|_F^2 \quad (3)$$

Pairwise matching constraint

The pairwise matching model has been proposed in learning to ranking literature [10]. It is the crucial component of the PSES model which accounts for document ranking. Following this methodology, we model document ranking in the pairwise style where the relevance information is in the form of preferences between pairs of documents with respect to individual queries. The constraint asks relevant query-document pairs to be close and irrelevant pairs to be far away in the semantic-rich embedding spaces of documents and queries.

Let us assume that the query x_q has a relevant document x_{d+} and an irrelevant document x_{d-}. In practice, x_{d+} is chosen according to annotated query-document pairs that are obtained from click-through data or pseudo-labeled data. x_{d-} is randomly selected from the document collection. The constraint forces the latent representation of the document x_{d+} to be near to the latent representation of the query x_q, and meanwhile the embedding vector of the document x_{d-} to be far from the embedding vector of x_q. We follow previous studies such as [7] and empirically find that using cosine similarity as the distance measure of latent representation leads to satisfactory performance in our data set. With such a measure, the pairwise matching loss L_{par} on the triplet set QD can be defined with hinge loss as:

$$L_{par} = \sum_{(q,d+,d-)\in QD} max[0, \beta - (sim(x_q, x_{d+}) - sim(x_q, x_{d-}))] \quad (4)$$

where β is the hyper-parameter.

Reconstruction constraint

The use of reconstruction constraint in our model guarantees that information can be recovered after feature space splitting. As in figure 1, the reconstruction loss L_{con} on a query set Q and a document set D can be defined in squared L2 norm as:

$$L_{con} = \sum_{q\in Q} \|x_{rq} - x_q\|_2^2 + \sum_{d\in D} \|x_{rd} - x_d\|_2^2 \quad (5)$$

where x_{rq} and x_{rd} are obtained from the decoding networks NN_{rq} and NN_{rd} respectively.

Adversarial learning

The process of learning the optimal feature representation should be conducted by jointly minimizing the adversarial loss in equation 2 and the other three losses in equations 3, 4, 5. We obtain the overall loss function L to optimize by putting the above constraints together, which is:

$$L = \gamma_1 \cdot L_{par} + \gamma_2 \cdot L_{ort} + \gamma_3 \cdot L_{con} + \gamma_4 \cdot L_{reg} + L_{adv} \quad (6)$$

where L_{reg} is the regularization term defined as sum of Frobenius norm of all weight matrices, and $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ are hyper-parameters. The competing optimization process can be implemented with mini-batch gradient descent and we use Adam [8] to compute the gradient.

4 EXPERIMENTS AND RESULTS

In this section, we conduct IR experiments to compare our model with traditional IR models and several neural IR models.

IR collections. For retrieval experiments, we make use of TREC collections consisting of different sizes and genres of heterogeneous text collections, which are one Robust track and one Web track. These corpora have been broadly used in previous IR literature. The characteristics of these document sets and corresponding queries are given in table 1. Since the ClueWeb-09-Cat-B collection (*ClueWeb* for short) is noisy, we filter the documents with spam scores in the 60-th percentile with Waterloo Fusion spam scores. For TREC queries, we make use of title fields for retrieval.

Table 1: TREC collections (M = million, B=Billion).

Data set	# Doc	# Word	Topic ID
Robust04	0.5M	252M	301-450, 601-700
ClueWeb	34.0M	26.1B	1-200

Training set. To train PSES, we choose to combine relevance signals from both a broadly used data set guaranteeing high quality and unsupervised IR models producing data in larger volume [4]. These two resources complement each other. The former data is directly picked from the LETOR4.0 [13] dataset which is designed to evaluate learning to rank models. In order to build the latter data with diversity, we combine as training queries a sample from AOL queries [12] and titles of new pages (~2.8M) crawled from several news sites such as ChinaDaily and XinhuaNews. These queries are utilized to retrieve the document collection with the unsupervised BM25 model. For each training query, we take the top 500 retrieved documents as positive samples. The negative samples are picked randomly from the document collection. We use pseudo-labeled data to train the whole PSES model prior to fine-tuning NN_s with the LETOR data set.

Experimental setup. In the PSES model, the input features are constructed based on the 512d word2vec embeddings trained on Wikipedia dump. The networks NN_q, NN_d, NN_s, NN_{rq}, NN_{rd} are all implemented as feed-forward networks of which the hidden layer numbers are chosen from $\{1, 2, 3\}$. The latent layer size is gotten from $\{64, 128, 256, 512, 1024\}$. The adversarial network NN_{adv} is a two-layer feed-forward network with softmax layer on top of the last layer. The hyper-parameters $\gamma_1, \gamma_2, \gamma_3, \gamma_4$ are chosen from $\{0.01, 0.1, 1, 10, 100\}$. The initial learning rate is selected from $\{10^{-3}, 10^{-4}, 5 * 10^{-4}, 10^{-5}, 5 * 10^{-5}\}$ and the batch size is fixed to 128. Those hyper-parameters are tuned on the validation set (20% of the training queries used for validation).

Performance measures. In order to measure the retrieval performance, we use mean average precision (MAP), precision at rank 20 (P20), and normalized discounted cumulative gain at rank 20 (nDCG). Statistically significant differences are determined using the two-tailed paired t-test with $p < 0.05$.

Baseline approaches. We compare the retrieval performance of our model PSES (M_P) with two categories of IR models: classic IR models showing state-of-the-art performance, and several neural IR models. We note that PSES is a semantic-based model rather than an interaction-focused model. As we have discussed in section 2,

we will pay more attention to the comparisons with semantic-based neural models, especially with those models designed to exploit pseudo-labeled data. The baseline models in comparison are:

Classic IR models: BM25 (M_B) model and query likelihood (QL) (M_Q) model with Dirichlet smoothing.

DSSM (M_D): A popularly cited neural model proposed in [7].

NRMS (M_N): A recently proposed neural model learned with weak supervision [4].

Table 2: Retrieval performance of all models on TREC collections. Significant improvement or degradation with respect to BM25 (M_B) is indicated by +/-.

	Robust04			ClueWeb		
	MAP	P20	nDCG	MAP	P20	nDCG
M_B	0.248	0.351	0.406	0.091	0.237	0.190
M_Q	0.245	0.352	0.404	0.092	0.239	0.193
M_D	0.227^-	0.315^-	0.382^-	0.061^-	0.221^-	0.176^-
M_N	0.280^+	0.383^+	0.443^+	0.129^+	0.301^+	0.237^+
M_P	$\mathbf{0.296}^+$	$\mathbf{0.403}^+$	$\mathbf{0.455}^+$	$\mathbf{0.144}^+$	$\mathbf{0.320}^+$	$\mathbf{0.259}^+$
M_{Pa}	0.283^+	0.389^+	0.442^+	0.132^+	0.308^+	0.243^+
M_{Pb}	0.271^+	0.374^+	0.438^+	0.125^+	0.306^+	0.231^+
M_{Pc}	0.270^+	0.367^+	0.429^+	0.118^+	0.300^+	0.231^+

Comparisons to state-of-the-art. Table 2 (except the last three rows) reports the experimental results on the TREC datasets. We firstly compare all baseline models with BM25, the one that has been used to construct pseudo-labeled data. The neural IR model DSSM performs significantly worse than BM25, which is coincident with findings in existing literature. NRMS is a neural ranking model trained with pseudo-labeled data, which significantly outperforms BM25 in all cases. The PSES model proposed in this paper, by splitting the embedding space into shared and query/document specific spaces, achieves the best overall performance. In fact, PSES always significantly outperforms BM25 by a large margin. We further compare PSES with NMRS and find that our model PSES is significantly better than NRMS on both collections considered in our experiments. We can draw the empirical conclusion that PSES is significantly better than traditional IR models and typical neural IR models in the same category.

Variants of PSES. Our PSES model can be altered in different ways. We consider here three variants: (1) M_{Pa} that is obtained by removing the orthogonality constraint from PSES. (2) M_{Pb} that is obtained by removing the adversarial constraint from PSES. (3) M_{Pc} that is obtained by removing both orthogonality and adversarial constraints from PSES. The experimental results are listed in the last three rows of table 2. We find from the results that all the variants degrade from M_P with significance. More precisely, removing the adversarial constraint affects the retrieval performance more than removing the orthogonality constraint. The worst performance is obtained when both constraints are removed. It supports our claim that different embedding spaces overlap with each other and separating them helps the retrieval task.

5 CONCLUSIONS

In this paper, we propose a novel representation learning model for IR which makes use of partially shared embedding spaces. As we can tell, it is the first time that queries and documents are encoded into different but overlapping subspace in IR. Experiments on TREC collections show that our model is significantly better than traditional IR models and typical neural models in the same category. A future direction to investigate is to combine PSES with interaction-based models to gain further improvement.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their valuable comments. This work was supported by the Fundamental Research Funds for Central Universities of CCNU (No. CCNU15A05062).

REFERENCES

[1] Martin Arjovsky, Soumith Chintala, and Léon Bottou. 2017. Wasserstein Generative Adversarial Networks. In *Proceedings of the 34th International Conference on Machine Learning (ICML)*. 214–223.

[2] Konstantinos Bousmalis, George Trigeorgis, Nathan Silberman, Dilip Krishnan, and Dumitru Erhan. 2016. Domain Separation Networks. In *Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS)*. 343–351.

[3] Stephane Clinchant and Florent Perronnin. 2013. Aggregating Continuous Word Embeddings for Information Retrieval. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*. 100–109.

[4] Mostafa Dehghani, Hamed Zamani, Aliaksei Severyn, Jaap Kamps, and W. Bruce Croft. 2017. Neural Ranking Models with Weak Supervision. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 65–74.

[5] Debasis Ganguly, Dwaipayan Roy, Mandar Mitra, and Gareth J.F. Jones. 2015. Word Embedding Based Generalized Language Model for Information Retrieval. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 795–798.

[6] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative Adversarial Nets. In *Proceedings of the 27th International Conference on Neural Information Processing Systems (NIPS)*. 2672–2680.

[7] Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry Heck. 2013. Learning Deep Structured Semantic Models for Web Search Using Clickthrough Data. In *Proceedings of the 22nd ACM International Conference on Information and Knowledge Management (CIKM)*. 2333–2338.

[8] Diederik P. Kingma and Jimmy Ba. 2015. Adam: A Method for Stochastic Optimization. In *Proceedings of the 3rd International Conference on Learning Representations (ICLR)*.

[9] Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. 2017. Adversarial Multi-task Learning for Text Classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (ACL)*. 1–10.

[10] Tie-Yan Liu. 2009. Learning to Rank for Information Retrieval. *Foundations and Trends in Information Retrieval* 3, 3 (2009), 225–331.

[11] Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to Match Using Local and Distributed Representations of Text for Web Search. In *Proceedings of the 26th International Conference on World Wide Web (WWW)*. 1291–1299.

[12] Greg Pass, Abdur Chowdhury, and Cayley Torgeson. 2006. A Picture of Search. In *Proceedings of the 1st International Conference on Scalable Information Systems (InfoScale)*.

[13] Tao Qin and Tie-Yan Liu. 2013. Introducing LETOR 4.0 Datasets. *CoRR* abs/1306.2597 (2013). arXiv:1306.2597

[14] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. A Latent Semantic Model with Convolutional-Pooling Structure for Information Retrieval. In *Proceedings of the 23rd ACM International Conference on Information and Knowledge Management (CIKM)*. 101–110.

[15] Yelong Shen, Xiaodong He, Jianfeng Gao, Li Deng, and Grégoire Mesnil. 2014. Learning Semantic Representations Using Convolutional Neural Networks for Web Search. In *Proceedings of the 23rd International Conference on World Wide Web (WWW)*. 373–374.

[16] Cedric Villani. 2008. *Optimal Transport: old and new*. Springer-Verlag.

[17] Hamed Zamani and W. Bruce Croft. 2016. Embedding-based Query Language Models. In *Proceedings of the 2016 ACM International Conference on the Theory of Information Retrieval (ICTIR)*. 147–156.

[18] Hamed Zamani and W. Bruce Croft. 2017. Relevance-based Word Embedding. In *Proceedings of the 40th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 505–514.

[19] Guoqing Zheng and Jamie Callan. 2015. Learning to Reweight Terms with Distributed Representations. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR)*. 575–584.