TITA: A Two-stage Interaction and Topic-Aware Text Matching Model

Xingwu Sun[1*]**,Yanling Cui**[2,3]**,Hongyin Tang**[2,3]**,Qiuyu Zhu**[1]**,Fuzheng Zhang**[1]**,Beihong Jin**[2,3*]
[1]Meituan Inc., Beijing, China
[2]State Key Laboratory of Computer Sciences, Institute of Software, Chinese Academy of Sciences
[3]University of Chinese Academy of Sciences, Beijing, China
*The corresponding authors: sunxingwu01@gmail.com, beihong@iscas.ac.cn

Abstract

In this paper, we focus on the problem of keyword and document matching by considering different relevance levels. In our recommendation system, different people follow different hot keywords with interest. We need to attach documents to each keyword and then distribute the documents to people who follow these keywords. The ideal documents should have the same topic with the keyword, which we call topic-aware relevance. In other words, topic-aware relevance documents are better than partially-relevance ones in this application. However, previous tasks never define topic-aware relevance clearly. To tackle this problem, we define a three-level relevance in keyword-document matching task: topic-aware relevance, partially-relevance and irrelevance. To capture the relevance between the short keyword and the document at above-mentioned three levels, we should not only combine the latent topic of the document with its deep neural representation, but also model complex interactions between the keyword and the document. To this end, we propose a Two-stage Interaction and Topic-Aware text matching model (TITA). In terms of "topic-aware", we introduce neural topic model to analyze the topic of the document and then use it to further encode the document. In terms of "two-stage interaction", we propose two successive stages to model complex interactions between the keyword and the document. Extensive experiments reveal that TITA outperforms other well-designed baselines and shows excellent performance in our recommendation system.

1 Introduction

The keyword-document matching is mostly like the query-document matching task. The query-document matching task, aiming to calculate relevance score between a query and a document, has been extensively studied over the past few years. It is widely applicable in many real scenarios: (1) in the information retrieval systems (Guo et al., 2016), query-document matching is an important feature in the ranking models. (2) as for the task of question answering (Yang et al., 2016), query-document matching method can be used to find document candidates or to help predict the answer span. (3) it is also widely applied to recommendation systems (Jiang et al., 2019).

In many scenarios, we need to distinguish different keyword-document (query-document) relevance levels. For instance, in our recommendation system, we need to attach documents to some hot keywords and then distribute the documents to the people who follow the keywords. In this circumstance, the document and the keyword should better have the same topic, which we call topic-aware relevance. As shown in Table 1, for the hot keyword "cherry blossoms", the document (labeled 2) should be the ideal document which should be attached because it has the same topic with the keyword while the document (labeled 1) should be a secondary choice, because only several words or phrases in this document match the keyword but the topics of the document mismatch the keyword.

To tackle this problem, we define a three-level relevance: topic-aware relevance, partially-relevance and irrelevance. The topic-aware relevance means the keyword and the document have the same topic while the partially-relevance means only part of the document matches with the keyword. Our task is more challenging than previous query-document matching tasks. To capture the relevance between the keyword and the document at above-mentioned three levels, we should not only combine the latent topic of the document with its deep representation, but also model complex interactions between the keyword and the document.

Previous neural query-document matching models (similar as keyword-document matching) can be divided into two categories according to their model architectures (Guo et al., 2016). One is the

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 5431–5440
June 6–11, 2021. ©2021 Association for Computational Linguistics

Keyword: cherry blossoms
Original Keyword: 樱花

Label: 0
Irrelevance Case
Translated Document: There was a flower shop which has opened for a few months. I bought some flowers to decorate my house. The shop had common flowers such as lilies and carnations, but there were not many colors to be chosen...
Original Document: 这家花店开了有几个月了。我买了一些花来装饰我的房子。店里有百合花、康乃馨等普通花卉，但可供选择的颜色不多...

Label: 1
Partially-Relevance Case
Translated Document: The food in this restaurant is very delicious. I tried some dishes, such as foie gras, steak, squid, noodles, desserts, etc. All the dishes are really yummy, especially the filet mignon... By the way, **there is a cherry blossoms exhibition near this restaurant.**
Original Document: 这家餐馆的菜都很好吃，我试吃了一些菜，如鹅肝、牛排、鱿鱼、面条、甜点等。所有的菜都很好吃，尤其是菲力牛排...顺便说一下，这家餐厅附近有樱花展。

Label: 2
Topic-Aware Relevance Case
Translated Document: **Yuyuantan Park is the best to enjoy cherry blossoms. The cherry blossoms in the park are available in a variety of colors and varieties. Their flowering period is short...**
Original Document: 玉渊潭公园是赏樱花的最佳去处。公园里的樱花有各种颜色和品种。它们的花期很短...

Table 1: A piece of example describing three levels of keyword-document relevance: topic-aware relevance, partially-relevance and irrelevance, which are labeled 2, 1 and 0 respectively. As for the keyword "cherry blossoms", the topic-aware relevance case and the partially-relevance case both have some words relevant to the keyword. However, the document, labeled 2, has the same topic with the keyword. By contrast, the topic of the partially-relevance document, labeled 1, is "restaurant", which mismatches the keyword. Note that this case is translated from Chinese.

representation-based models, in which representations for a query and a document are built independently. In other words, there are no word-level or phrase-level interactions between the query and the document. For instance, the well-known DSSM (Huang et al., 2013) has been verified effective in query-document matching tasks. However, these representation-based series cannot model complex interactive signals between a query and a document effectively. The other one we call interaction-based models, in which word or phrase-level information fusion occurs. It has been verified more effective to directly learn interactions than individual representations. Examples include ARC II (Hu et al., 2014), MatchPyramid (Pang et al., 2016). Recently, interaction-based methods are widely used in many NLP tasks, like BIDAF (Seo et al., 2016) and R-NET (Wang et al., 2017).

More recently, BERT (Devlin et al., 2018) has made great influence in the field of NLP. It has achieved state-of-the art results in many NLP applications. The pre-trained language models can be applied directly to this keyword-document matching task.

However, these above-mentioned types of keyword-document (query-document) matching models can be improved to be applied to our recommendation system in the following aspects: (1) They do not analyse the topic of the document. It is expected that topic model can be used to solve this problem. (2) Previous interaction-based models can still be improved to capture complex matching signals between a query and a document. To this end, we propose the TITA model. By topic-aware, we introduce neural topic model (Miao et al., 2017) to analyze the latent topic representation of the document and then use this latent topic to further encode the document. By two-stage interaction, we propose a two-stage interaction to model complex interactions between a query and a document.

Our research contributions can be summarized as follows.

- We observe two major shortcomings in current keyword-document matching models and propose the TITA model to improve them. Our model has two advantages: (1) it encodes the latent topic embedding into the deep neural representation of the document, which can aid the prediction of the topic-aware relevance. (2) it can model more complex interactions between a keyword and a document through a two-stage keyword-document interaction.

- We perform extensive experiments on our keyword-document matching dataset. The results reveal that the proposed TITA model outperforms the well-designed baselines.
- From a real recommendation system, we define a three-level relevance in keyword-document matching task and construct a new dataset.
- Our model is applied in our recommendation system and improves the click-through rate by 4.35%.

2 Related Work

Depending on the model architectures, text matching models can be divided into two categories: representation-based and interaction-based. The former ones first transform every piece of text to a representation with neural networks, such as Deep Semantic Similarity Model(DSSM) (Huang et al., 2013), Convolutional Deep Semantic Similarity Model(CDSSM) (Shen et al., 2014), LSTM-RNN (Palangi et al., 2016), Bi-LSTM, etc. Conversely, the latter models focus on modeling the interaction between a query and a document, such as Arc-II(Hu et al., 2014), MatchPyramid (Pang et al., 2016), BIDAF(Seo et al., 2016) and RNET(Wang et al., 2017).

Representation-based methods generate distributed representations from input texts through neural networks. There are a number of works employing these methods, which differ mainly in the procedure to construct the representations and the way of calculating a matching score. Huang et al. (2013) propose DSSM, which is the first one to apply a neural network. In DSSM, each piece of the query or the document is represented through a multilayer perceptron and then a matching score is calculated by the cosine similarity. Compared to traditional text matching models, DSSM shows significant improvements.

Compared with representation-based methods, the interaction-based methods aim to capture direct matching features: the degree and the structure of matching. The interaction-based model, which means query-document interaction occurs before matching, can somewhat solve the above-mentioned problem in the representation-based models. It has been verified more effective to directly learn interactions than individual representations. Hu et al. (2014) propose ARC-II, which first represents the query and the document by the knowledge of each other, and adjusts the sliding windows in the first convolution layer to focus on adjacent word vectors. Inspired by the success of convolutional neural network in image recognition, Pang et al. (2016) propose MatchPyramid to model text matching as the problem of image recognition. Leveraging the attention mechanism, Seo et al. (2016) and Wang et al. (2017) introduce attention mechanism to improve the matching degree of the query and the document.

Recently BERT (Devlin et al., 2018) has caused a stir in the field of NLP. It has achieved state-of-the-art results in many NLP applications. The pre-trained language model series can be applied directly to this keyword-document matching task.

Topic models aim to discover the topics as well as the topic representations of documents in the document collection. It learns latent topics from documents in an unsupervised manner. Topics are captured as latent variables that have a word probability distribution. Topic models have a long tradition in this scenario area as well, such as bibliometrics, translations and recommendations.

Hall et al. (2008) describe the flow of topics between papers. Zhao and Xing (2006) enable word alignment process to leverage topical contents of document-pairs. Jiang et al. (2015) use topic model to enrich users' information for effective inference.

3 Our Model

In this section, we describe details of the TITA model. As depicted in Figure 1, our TITA model has three major components: (1) a two-stage keyword-document interaction, see Part A; (2) a neural topic model, see Part B; (3) a joint training mechanism, see Part C. First, we introduce the task definition. Then, we elaborate the two-stage keyword-document interaction and neural topic model in the TITA model respectively. Finally, a joint training mechanism is introduced to incorporate latent topics to the deep representation

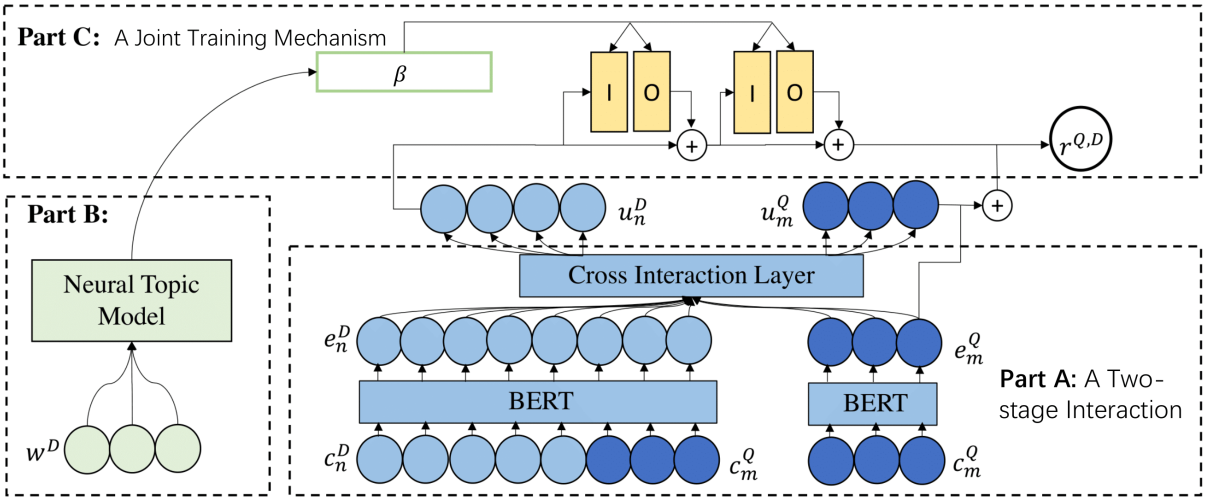

Figure 1: The architecture of the TITA model, which consists of three major components: (1) a two-stage keyword-document interaction, which combines the multi-head attention in BERT and a successive cross representation layer to link the keyword and the document; (2) a neural topic model, which calculate a latent topic of the document to further enrich the document representation; (3) a joint training mechanism to train the model in a joint process. In this part, "I" indicates the input memory while "O" indicates the output memory.

of the document and train the model in a joint process. Notably, we conduct experiments using both Bi-LSTM and BERT as text encoders. Here, we only describe the proposed methods with BERT as the encoder for simplicity.

3.1 Task Definition

In our keyword-document matching task, we explicitly model the relevance between a keyword and a document as a relevance level prediction task. The input of the task is a keyword Q and a document D. The output $r^{Q,D} \in \{0, 1, 2\}$ indicates the keyword-document relevance levels.

3.2 A Two-stage Keyword-document Interaction

The keyword-document matching model is desired to capture the rich interactions between the keyword and the document in the matching process. As show in Table 1, the keyword "cherry blossoms" and the topic-aware relevance document have many correlating signals, e.g., the phrase "cherry blossoms" in the keyword and the phrase "flowering period" in the document.

The two-stage keyword-document interaction in the TITA model is to fuse the information of the document and the keyword. In the first-stage interaction, we employ BERT (Devlin et al., 2018) as the encoder to simultaneously model the sequential information of the keyword and the document along with their interactive relationship by the multi-head self-attention mechanism. In the second-stage interaction, we perform a cross-attention between the representations of the keyword and the document to further capture their interactive relationship.

First-stage Interaction As shown in Figure 1, in the first-stage interaction, we concatenate the keyword and the document by a separator [SEP] as input and then feed them into BERT. The input consists of the keyword characters $c^Q = \{c_m^Q\}_{m=1}^M$ and the document characters $c^D = \{c_n^D\}_{n=1}^N$, where M, N indicate the length of the keyword characters and the document characters respectively. The states in the last hidden layer of BERT can be regarded as the encoding of the document, i.e., e^D.

$$e^D = \text{BERT}([c^Q; [\text{SEP}]; c^D]) \qquad (1)$$

where $e^D = \{e_n^D\}_{n=1}^N \in \mathbb{R}^{N \times d}$. In each hidden layer of BERT, the multi-head self-attention mechanism is performed as the following equations:

$$\text{Attention}(\tilde{Q}, \tilde{K}, \tilde{V}) = \text{softmax}(\frac{\tilde{Q}\tilde{K}^T}{\sqrt{d_k}})\tilde{V} \qquad (2)$$

$$\text{MultiHead}(\tilde{Q}, \tilde{K}, \tilde{V}) = \text{Concat}(\text{hd}_1, ..., \text{hd}_h)W^O \qquad (3)$$

$$\text{hd}_i = \text{Attention}(\tilde{Q}W_i^{\tilde{Q}}, \tilde{K}W_i^{\tilde{K}}, \tilde{V}W_i^{\tilde{V}}) \qquad (4)$$

where $\tilde{Q}$, $\tilde{K}$ and $\tilde{V}$ are the output hidden states of the former layer. $W_i^{\tilde{Q}}$, $W_i^{\tilde{K}}$ and $W_i^{\tilde{V}}$ are the parameters corresponding to each head. W^O is the output projection parameter. For more details, readers can refer to (Devlin et al., 2018).

Second-stage Interaction Note that in the first-stage interaction, the query and the document characters are concatenated as input. The model learns keyword-keyword, keyword-document, document-document interactions simultaneously through self-attention mechanism in transformer blocks of BERT. In our keyword-document matching task, keyword-document interaction is more important than document-document and keyword-keyword interactions. Therefore, we introduce the second-stage interaction layer to conduct keyword-document contextualization independently. Firstly, we obtain the representation of the keyword e^Q by the BERT encoder.

$$e^Q = \text{BERT}(c^Q) \in \mathbb{R}^{M \times d} \tag{5}$$

where $e^Q = \{e^Q_m\}^M_{m=1}$. Then, we compute a similarity matrix using the keyword embedding and the document embedding.

$$S = (s_{mn}) \in \mathbb{R}^{M \times N} \tag{6}$$

$$s_{mn} = \left\langle e^Q_m, e^D_n \right\rangle v^T \in \mathbb{R} \tag{7}$$

where $\langle e^Q_m, e^D_n \rangle$ represents a element-wise multiplication, $v \in \mathbb{R}^d$ is a trainable weight vector. In this similarity matrix, the value s_{mn} indicates the link between the m-th character embedding in the keyword and the representation of the n-th character in the document. Then, we apply this similarity matrix to further encode the keyword by calculating attention over the document:

$$u^Q = \{u^Q_m\}^M_{m=1} \tag{8}$$

$$u^Q_m = \sum_{n=1}^{N} a_{mn} e^D_n \in \mathbb{R}^d \tag{9}$$

$$a_m = \text{softmax}(s_m) \in \mathbb{R}^N \tag{10}$$

where $s_m = \{s_{mn}\}^N_{n=1}$ and a_m means which characters in the document should be attended regarding the m-th character of the keyword. We then add the original keyword representation e^Q with u^Q to get the keyword embedding:

$$u^Q = u^Q + e^Q \tag{11}$$

Similarly, we use this similarity matrix to get the document representation $u^D \in \mathbb{R}^{N \times d}$.

3.3 Neural Topic Model

As show in Table 1, the topic-aware relevance case and the partially-relevance case both have some words relevant to "cherry blossoms". But the topic of the topic-aware relevance document is more related with the keyword "cherry bollosoms". By contrast, the topic of the partial-relevance document is more likely to be a document about a "restaurant", which is not related to the keyword "cherry blossom". Following this direction, analysing the topic of the document is a way to promote keyword-document matching models. Specifically, we introduce neural topic model to produce the latent topic and then use it to update the upstream representation of the document.

As shown in Figure 1, the input of the neural topic model is a word sequence of the document w^D. The bag-of-words (BOW) representation of the document is $x^D \in \mathbb{R}^{|V_w|}$, where $|V_w|$ is the size of the word vocabulary. Assume that the latent variable θ represents the topic distribution in the document w^D. The probabilistic topic models, like LDA(Blei et al., 2003), apply the Dirichlet distribution as the prior of the latent variable $\theta \sim Dir(\alpha)$, where α is the parameter of the Dirichlet distribution. By contrast, in the neural topic model, Gaussian Softmax Construction (Miao et al., 2017) is applied using a neural network to parameterise the topic distribution $\theta \sim G_{GSM}(\mu_0, \sigma_0^2)$:

$$x \sim \mathcal{N}(\mu_0, \sigma_0^2) \tag{12}$$

$$\theta = \text{softmax}(W_1^T x) \tag{13}$$

where W_1 is a trainable parameter. μ_0 and σ_0 are the parameters of the prior Gaussian distribution $\mathcal{N}$. Assuming there are K topics, if $z_n \in \{1, ..., K\}$ is the topic assignment for the observed word w^D_n, then:

$$z_n \sim \text{Multi}(\theta) \tag{14}$$

$\beta_{z_n} \in \mathbb{R}^{|V_w|}$ is a topic distribution over the words in the vocabulary given z_n. The topic distribution can be calculated by the similarity between the topic and the words in the vocabulary:

$$\beta_{z_n} = \text{softmax}(\tilde{v}^T t_{z_n}) \tag{15}$$

where $t \in \mathbb{R}^{d \times K}$ is the topic vector which is a parameter of the neural topic model, $\tilde{v} \in \mathbb{R}^{d \times |V_w|}$ is the word vector. K is the total topic number. Then, the generative probability of each word w_n

can be calculated by:

$$p(w_n|\beta_{z_n}) = \text{Multi}(\beta_{z_n}) \tag{16}$$

The neural topic model is implemented by an Auto-Encoding Variational Bayes (AEVB) algorithm (Kingma and Welling, 2013). The encoder is used to approximate the true posterior of the latent variable $p(\theta|x)$. Specifically, the encoder takes the BOW (Bag-of-Words) representation of the document as the input and generates the posterior Gaussian Softmax Construction parameters μ and σ^2 through neural networks. In practice, the latent variable θ is sampled by the reparameterization trick.

$$\mu = f_1(x^D), \log\sigma = f_2(x^D) \tag{17}$$

$$\theta \sim G_{GSM}(\mu, \sigma^2) \tag{18}$$

where $f_*(\cdot)$ is a multi-layer perceptron. The decoder is responsible for reconstructing the document by maximizing the log likelihood of the input document. The latent variable z_n can be integrated out as follows.

$$\log p(w_n|\beta,\theta) = \log \sum_{z_n} [p(w_n|\beta_{z_n})p(z_n|\theta)] \tag{19}$$

$$= \log(\theta \cdot \beta) \tag{20}$$

Finally, the variational lower bound of the neural topic model is obtained by combining the reconstruction error term and the KL divergence term. The parameters of neural topic model can be trained by maximizing this function.

$$\begin{aligned}\mathcal{L}_{NTM} =& \mathbb{E}_{p'(\theta|D)}\left[\sum_{n=1}^{N} log \sum_{z_n}[p(w_n|\beta_{z_n})p(z_n|\theta)]\right] \\ &- D_{KL}[p'(\theta|D)||p(\theta|\mu_0, \sigma_0^2)]\end{aligned} \tag{21}$$

where $p'(\theta|D)$ means the variational posterior distribution of document D, approximating the true posterior $p(\theta|D)$.

3.4 A Joint Training Mechanism

It's expected that introducing topic model can benefit the model in the prediction of the above-mentioned three levels. In this subsection, we design a joint training mechanism to incorporate the latent topic representation to further encode the document and train the model in a joint process.

As described above, u^D is the document representation after the two-stage keyword-document interaction. $\beta \in R^{K\times|V_w|}$ is the topic distribution over the vocabulary, where β_{ij} means that the weight between the i-th topic and the j-th word. We are inspired from an end2end memory network(Sukhbaatar et al., 2015), which is used to memorize multiple sentences in question answering task. Similarly, in TITA, we intend to embed the topic-word weight into the deep representation of the document.

As depicted in Figure 1 part C, the input of memory network is β and the deep document representation after the two-stage keyword-document interaction u^D. β is memorized in the memory of the network, where β_k means the representation of the k-th topic over the vocabulary of size $|V_w|$.

The TITA model has two memory hops as shown in the Figure 1. In the following, we describe the model in a single memory hop operation for simplicity. One hop has two major components: the input memory and the output memory. In the input memory representation, a matching score is calculated taking β and u^D as input:

$$p_k = \text{softmax}(\beta_k V u^D) \tag{22}$$

where $V \in R^{|V_w|\times d}$ is a trainable weight vector. In the output memory representation part, we compute the slot output vector using the output memory and the matching score:

$$o^D = \sum_{k=1}^{K}(p_k c_k) \tag{23}$$

$$o^D = W_o(o^D + u^D) \tag{24}$$

where $c \in \mathbb{R}^{K\times d}$ is a trainable output memory. We compute two relevance vectors r_1 and r_2. One takes u^D and u^Q as input, while the other one using u^Q and o^D. We merge the two relevance vectors and then apply softmax function to get the final relevance level:

$$r_1 = W_{R1}[u^D; u^Q] + b_{R1} \tag{25}$$

$$r_2 = W_{R2}[o^D; u^Q] + b_{R2} \tag{26}$$

$$r^{Q,D} = \text{softmax}(W_R[r_1; r_2] + b_R) \tag{27}$$

where $[;]$ is vector concatenation operation and W_*, b_* are all trainable variables.

The TITA model integrates three different parts as shown in Figure 1: a two-stage keyword-document interaction, a neural topic model and a joint training mechanism. In the training process, the neural topic model and the joint model are trained alternatively to a convergent status. We first train the neural topic model for λ epochs to get a topic distribution over vocabulary, i.e., β. Then in the joint training process, the model takes the output of the two-stage keyword-document interaction and the output of the neural topic model to conduct training the model parameters for classification.

4 Experiment

In this section, we conduct experiments on our keyword-document matching dataset from our recommendation system and the results demonstrate the superiority of the TITA model compared to the baselines.

We apply accuracy as the evaluation metric. In this paper, we care mostly about rigidly distinguishing the three keyword-document matching levels. We believe that documents of different matching levels have different usages. For instance, in our online recommendation system, the goal of our model is to recall the topic-aware relevance documents and there is no need to rank documents of each keyword.

4.1 Dataset

Our keyword-document matching dataset is in Chinese, derived from our recommendation system. The domains mainly lie in food (e.g., beef and western food), sports (e.g., football and jogging), entertainment (e.g., photography and comedy) and so on. For all the 8901 keywords, we get 10 documents for each keyword by users' behavior in our recommendation system, e.g., click-through. As for how to choose 10 documents for each keyword in the baseline online recommendation system, for a certain keyword, hundreds of documents are recalled for different users, in which the topic-aware relevance documents tend to have high click-through rate while irrelevance ones tend to have low click-through rate. For each keyword, we select 6 documents which have high click-through rate as well as 4 documents with low click-through rate. According to our analysis, this setting tends to generate similar ratios of three-level relevance documents for all the keywords. As a result, each keyword has 10 corresponding documents. Each keyword-document pair is manually annotated at different relevance levels. As shown in Table 1, relevance level-2 means the document and the keyword have the same topic, while relevance level-0 means the keyword and the document are irrelevant. Relevance level-1 is an intermediate relevance level, which means only a small portion of the document describes some useful information of the keyword. To make the ratios of level-2, level-1, level-0 cases nearly the same, we randomly delete some documents. As a result, we have 8,901 keywords and 66,019 corresponding documents. Finally, the dataset is randomly split into 50% for training, 25% for validation and 25% for testing.

4.2 Experiment Settings

In the experiment, we set the cutoff length of the document sequence as 512 characters and the cutoff length of the keyword as 16 characters in Chinese. The size of the character vocabulary V_c is 21128. The size of the word vocabulary for neural topic model V_w is 5000, which contains top frequent words after deleting stop words. We use pre-trained embeddings by BERT to initialize the character embeddings. We directly use BERT base model released by Google with the hidden size of 768. In the neural topic model, we set the number of topics $\#K = 50$. We use all the documents in the training set to train the neural topic model for 50 epochs. The topic embedding size d is set to 384 and we set the word embedding to the same size. The padding is masked to avoid affecting the gradient. We use the optimization algorithm Adam (Kingma and Ba, 2014) with learning rate 5e-5 and batch size as 32. As for the parameters of Adam, β_1 and β_2 are set to 0.9 and 0.999 respectively.

4.3 Baselines

As described in the Introduction Section, the keyword-document matching models can be divided into two categories: representation-based and interaction-based matching model. As shown in Table 2, many strong baselines are included in the performance comparison.

4.4 Main Results and Ablation Analysis

Table 2 shows that the TITA model outperforms all the models evaluated by accuracy in this keyword-document matching task. From this table, we have the other observations: (1) The TITA model is more competent in this task. It outperforms ARC-II by 7.06% and outperforms BERT by 5.38%, which

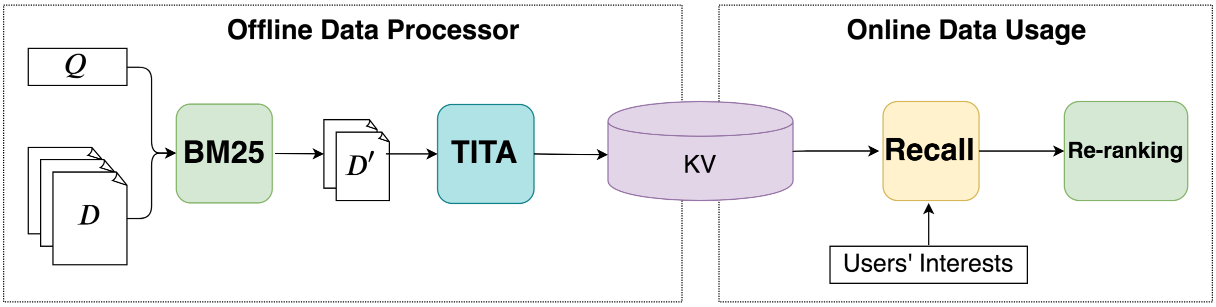

Figure 2: The architecture of online deployment of the TITA model, which consists of two major components: an offline data processor module and an online data usage module.

Models	Acc(%)
Bi-LSTM	67.16
DSSM (Huang et al., 2013)	68.66
CLSM (Shen et al., 2014)	67.21
DSSM-LSTM (Palangi et al., 2016)	66.71
MatchPyramid (Pang et al., 2016)	66.89
ARC-II (Hu et al., 2014)	68.16
BIDAF (Seo et al., 2016)	67.55
RNET (Wang et al., 2017)	67.69
BERT (Devlin et al., 2018)	69.84
The TITA Model	**75.22**

Table 2: The main experimental results of baselines and the TITA model evaluated by accuracy.

Models	Acc(%)
Bi-LSTM	67.16
+ Neural Topic Model	**69.18**
+ First-stage Keyword-document Interaction	70.86
+ Second-stage Keyword-document Interaction	72.45
Replace Bi-LSTM with BERT	**75.22**

Table 3: Ablation test results of the TITA model evaluated by accuracy.

strongly proves that topic model and two-stage interaction can benefit this task. (2) Most interaction-based models behave better than representation-based ones. (3) Pre-trained word embeddings can also aid this task.

To further examine the effectiveness of the neural topic model and the two-stage keyword-document interaction, we make a detailed ablation analysis as shown in Table 3.

- **Bi-LSTM**: The TITA model is based on Bi-LSTM, which encodes a query and a document independently before matching.

- **+ Neural Topic Model**: Bi-LSTM plus neural topic model outperforms the Bi-LSTM baseline by a large scale (i.e., 2.02%), which indicates that the keyword-document matching task can benefit from the latent topic representation of the document.

- **+ First-stage Keyword-document Interaction**: After adding the first-stage keyword-document interaction, the model behaves better. It proves that concatenating the query and document to conduct interaction is effective.

- **+ Second-stage Keyword-document Interaction**: We add the second-stage interaction to make further improvement. We infer that the cross attention is more capable in capturing interactions between a keyword and a document.

- **Replace Bi-LSTM with BERT**: We apply BERT to initialize the word representation, whose parameters are to be finetuned. We can observe that the model performs even better than the former one, which reveals that the pre-trained word representations are useful in the keyword-document matching task.

5 Online Deployment and Online Gains

Because the model is heavy and the total numbers of keywords are limited (8901 in total), we generate data in offline, as shown in Figure 2. In offline data processor, we first use BM25 to retrieve and rank billions of document candidates and keep the top-10000 candidates for TITA model to further conduct query-document relation prediction. After that we can get a ranked list of topic matching documents and partially relevance documents for all keywords, which will be stored in a KV database. In the online data usage, we recall documents of all the keywords, which the user follows, for further re-ranking in our recommendation system.

As for the online gains, we attached more than one million topic-matching documents for the 8901 keywords. These documents are all distributed in our recommendation system with the number of views about $1.9e^6/day$. We improve the click-through rate by 4.35% (from 6.52% to 10.87%), which is a great improvement.

6 Conclusions

We define a new keyword-document matching task with three relevance levels from a real recommendation system, to address the problem that different scenarios require documents of different relevance levels. Further, we propose a TITA model to distinguish different relevance levels, which can capture latent topics of a document and hold complex keyword-document interactions at the same time. Extensive experiments reveal the superiority of our model compared to other strong baselines. Ablation test shows that the model can improve the keyword-document matching in the same way as we think. Moreover, our model shows excellent performance in our recommendation system, in which it improves the click-through rate by 4.35%.

References

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. volume 3, pages 993–1022.

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

Jiafeng Guo, Yixing Fan, Qingyao Ai, and W. Bruce Croft. 2016. A deep relevance matching model for ad-hoc retrieval. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, pages 55–64.

David Hall, Daniel Jurafsky, and Christopher D Manning. 2008. Studying the history of ideas using topic models. In *Proceedings of the conference on empirical methods in natural language processing*, pages 363–371. Association for Computational Linguistics.

Baotian Hu, Zhengdong Lu, Hang Li, and Qingcai Chen. 2014. Convolutional neural network architectures for matching natural language sentences. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2042–2050.

Po-Sen Huang, Xiaodong He, Jianfeng Gao, Li Deng, Alex Acero, and Larry P. Heck. 2013. Learning deep structured semantic models for web search using clickthrough data. In *22nd ACM International Conference on Information and Knowledge Management, CIKM'13, San Francisco, CA, USA, October 27 - November 1, 2013*, pages 2333–2338.

Jyun-Yu Jiang, Mingyang Zhang, Cheng Li, Michael Bendersky, Nadav Golbandi, and Marc Najork. 2019. Semantic text matching for long-form documents. In *The World Wide Web Conference, WWW 2019, San Francisco, CA, USA, May 13-17, 2019*, pages 795–806.

Shuhui Jiang, Xueming Qian, Jialie Shen, Yun Fu, and Tao Mei. 2015. Author topic model-based collaborative filtering for personalized poi recommendations. *IEEE transactions on multimedia*, 17(6):907–918.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational bayes.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, pages 2410–2419.

Hamid Palangi, Li Deng, Yelong Shen, Jianfeng Gao, Xiaodong He, Jianshu Chen, Xinying Song, and Rabab K. Ward. 2016. Deep sentence embedding using long short-term memory networks: Analysis and application to information retrieval. volume 24, pages 694–707.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA.*, pages 2793–2799.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

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 Conference on Information and Knowledge Management, CIKM 2014, Shanghai, China, November 3-7, 2014*, pages 101–110.

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198.

Liu Yang, Qingyao Ai, Jiafeng Guo, and W. Bruce Croft. 2016. anmm: Ranking short answer texts with attention-based neural matching model. In *Proceedings of the 25th ACM International Conference on Information and Knowledge Management, CIKM 2016, Indianapolis, IN, USA, October 24-28, 2016*, pages 287–296.

Bing Zhao and Eric P Xing. 2006. Bitam: Bilingual topic admixture models for word alignment. In *Proceedings of the COLING/ACL on Main conference poster sessions*, pages 969–976. Association for Computational Linguistics.