Focusing Attention Network for Answer Ranking

Yufei Xie
Qiniu AtLab
Shanghai, China
xieyufei@qiniu.com

Shuchun Liu
Qiniu AtLab
Shanghai, China
liushuchun@qiniu.com

Tangren Yao
Qiniu AtLab
Shanghai, China
yaotangren@qiniu.com

Yao Peng
Qiniu AtLab
Shanghai, China
pengyao@qiniu.com

Zhao Lu
East China Normal University
Shanghai, China
yufeixie@ica.stc.sh.cn

ABSTRACT

Answer ranking is an important task in Community Question Answering (CQA), by which "Good" answers should be ranked in the front of "Bad" or "Potentially Useful" answers. The state of the art is the attention-based classification framework that learns the mapping between the questions and the answers. However, we observe that existing attention-based methods perform poorly on complicated question-answer pairs. One major reason is that existing methods cannot get accurate alignments between questions and answers for such pairs. We call the phenomenon "attention divergence". In this paper, we propose a new attention mechanism, called **F**ocusing **A**ttention **N**etwork(FAN), which can automatically draw back the divergent attention by adding the semantic, and metadata features. Our Model can focus on the most important part of the sentence and therefore improve the answer ranking performance. Experimental results on the CQA dataset of SemEval-2016 and SemEval-2017 demonstrate that our method respectively attains 79.38 and 88.72 on MAP and outperforms the Top-1 system in the shared task by 0.19 and 0.29.

CCS CONCEPTS

• **Computing methodologies** → **Lexical semantics**; *Information extraction*; Discourse, dialogue and pragmatics.

KEYWORDS

Answer Ranking; Focusing Attention Network; Semantic Features; Metadata Features

ACM Reference Format:
Yufei Xie, Shuchun Liu, Tangren Yao, Yao Peng, and Zhao Lu. 2019. Focusing Attention Network for Answer Ranking. In *Proceedings of the 2019 World Wide Web Conference (WWW'19), May 13–17, 2019, San Francisco, CA, USA.* ACM, New York, NY, USA, 7 pages. https://doi.org/10.1145/3308558.3313518

WWW '19, May 13–17, 2019, San Francisco, CA, USA

ACM ISBN 978-1-4503-6674-8/19/05.
https://doi.org/10.1145/3308558.3313518

1 INTRODUCTION

Online community forums, such as Yahoo! Answers [1] and Stack Overflow [2], have attracted many users. These users can ask or answer questions, and these archived questions and answers are very valuable. However, some of these questions have many answers, and most of these answers are low quality. Thus, *answer selection* of community question answering, which could automatically find good ones from a large number of answers, has drawn much attention in the field of natural language processing [3, 28, 34]. There are related tasks at SemEval-2016 [1] and SemEval-2017 [18].

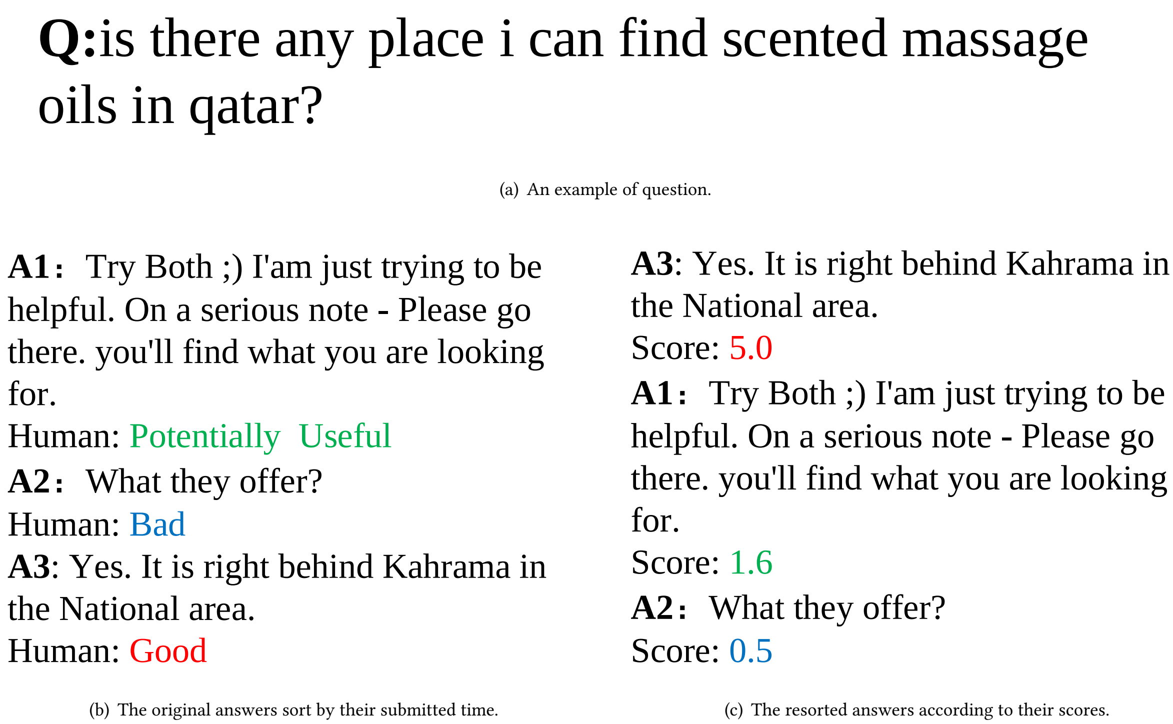
In this paper, we focus on the task of ranking good answers ahead of bad ones, which is another form of answer selection. We illustrate an example of Qatar-Living dataset [18] from the SemEval-2017 Task 3 in Figure 1. An original question Q is shown in Figure 1(a). For convenience, we only choose three of the ten answers of question Q and show in Figure 1(b). These answers are sorted by their submitted time. They have different human labels:"Good", "Potentially Useful" and "Bad". We need to train a model, which can give the score of each answer and the answers should be sorted by the score at last, as in Figure 1(c).

Previous work on answer ranking normally used feature engineering, linguistic tools, or external resources. For example, semantic features are based on WordNet in [34]. This model pairs semantically related words based on word semantic relations. In [28], the answer ranking problem is transformed to a syntactical matching between the question/answer parse trees.

In recent years, attention based models have shown advantages in representing natural language sentences [2]. It obtains great success in many NLP tasks such as machine translation, question answering and recognizing textual entailments. When building the representation of a sentence, some attention information is added to the hidden state. For example, in attention based recurrent neural networks models each time-step hidden representation is weighted by attention.

However, so far, these traditional attention based models were shown to be successful only in modeling relatively short query and answer pairs, roughly a sentence in length [22]. One of the reasons these models under-perform for complicated longer texts, is that longer answers often contain irrelevant information and the alignments estimated by the attention model are easily corrupted due to the complexity of sentences. In other words, the attention model

[1] https://answers.yahoo.com/
[2] http://stackoverflow.com/



(a) An example of question.

(b) The original answers sort by their submitted time.

(c) The resorted answers according to their scores.

Figure 1: An example question and its answers in CQA.

cannot accurately associate each question with the corresponding target region in the answers. We call this phenomenon *attention divergence*. That is the attention regions divergence in some degree from the proper regions of target terms in the answers. This motivates us to develop some mechanism to focus attention back on the right regions of the target terms in the answers. Therefore, we propose a Focusing Attention Network (**FAN**) to accurately identify the main question and answer terms to use for the joint representation by adding the semantic and metadata features. Semantic features, which can obtain the semantic similarities between a question and an answer; metadata features, which can capture characteristics that relevant answers tend to have. Both of them are very important features to represent the sentence and we call them focusing features.

We firstly obtain the question representation with bidirectional LSTMs, next we extract semantic features, metadata features from questions and their answers, then we obtain the attentive hidden states that are weighted by the question representation and above two kinds of features as answer representation. At last, we feed the question and answer representation to the softmax classifier for classification. The experimental results on SemEval-2016 Task3 and SemEval-2017 Task3 datasets show that our method outperforms the state-of-the-art work. The main contributions of our approach are three folds:

- We propose the concept of attention divergence, which explains the poor performance of existing attention based methods on complicated long question-answer pairs.
- We develop a novel method called **FAN** to solve the attention divergence problem, where in addition to the attention component existing in most existing methods, semantic features and metadata features is introduced, which can focus deviated attention back on the target areas.
- The experimental results have verified the effectiveness of our proposed model, and also provide a promising avenue to make full use of the **FAN** model for other tasks.

In the remainder of this paper, after discussing related work in Section 2, we introduce our model in Section 3. Then, we describe our experiments in Section 4. The conclusions and discussions are presented in Section 5.

2 RELATED WORK

Question-answer selection and ranking has been an active area of research for decades, presenting many solutions. For example, Filice [8] proposes a kernel-based method which uses the ranking features, heuristics features, thread-based features and stacking features. This method obtains the best result in SemEval-2017 Task 3 subtask A. The MAP value is 88.43. Feng [5] develops a ranking system based on traditional NLP features such as tf-idf cosine, longest common subsequence, word overlap and translation probability. This method also obtains a good result in SemEval-2017 Task 3 subtask A. The MAP value is 88.24. While these methods show effectiveness, they might ignore the sentence representation in the semantic space.

Most recently, various convolution neural networks (CNNs) and recurrent neural networks (RNNs) are employed in the task of answer ranking [20, 24, 26]. Neural networks based models are proposed to represent the sentence in a semantic space and then compare the question and answer candidates in this hidden space [26]. These models will not suffer from the availability of resources, the effort of feature engineering and the systematic complexity by introducing linguistic tools, such as parse trees and dependency trees [24]. Tan [24] utilizes a simple but efficient attention mechanism in order to generate the answer representation according to the question context. Wang [26] proposes an inner attention based recurrent neural network which adds attention information before RNN hidden representation and shows the advantage in representation. These methods have the characteristics of constructing relatively valuable vector representation and they improve the results in answer ranking task. However, these methods face the issue of information loss when obtaining the last output layer.

To tackle the issue of information loss, combining feature extraction and deep neural networks has attracted many researchers [12, 23, 35]. Wu [31] combines two different methods to represent question-comment pairs, i.e., supervised model using traditional features and convolutional neural network. Takahashi [23] proposes a deep neural network with acoustic features to solve the acoustic scene classification. Zhou [35] combines human engineered feature extraction as well as those learned by the deep neural network. Koreeda [12] proposes a method that combines neural network similarities and handcrafted comment plausibility features. Combining deep neural networks and traditional features have the ability of incorporating different kinds of information.

Different from the existing approaches, in this paper we introduce the semantic features and metadata features into the attention network, which can make the attention focus on the semantic and metadata of the sentences. Other than simply combines a traditional features and the output of the last layer of CNN [30].

Though attention divergence has been observed in attention training of speech recognition [11], where the authors proposed an MTL framework that combines CTC and attention network to handle this issue, our paper is the first work that formally puts forward the concept of attention divergence. Furthermore, we design a focus-mechanism to solve this problem in the answer ranking domain.

3 OUR MODEL

Figure 2 shows the architecture of our proposed FAN model. Specifically, we adopt the BiLSTM model to obtain the preliminary representations of the question and answer. To solve the problem of attention divergence, we extract semantic futures and metadata features to enrich the question representation and then obtain a focusing attention for answer representation. Finally, we employ the softmax classifier to make the classification. We describe the details of the model in this section.

3.1 Semantic Features

Semantic and metadata are the two important components of the sentence. In order to tackle the issue of attention divergences, semantic features are introduced into the task of answer ranking. We describe our approach of extracting semantic features from questions and their answers in details. In natural language processing (NLP), word embedding maps words or phrases from the vocabulary to vectors of real numbers. It involves a mathematical embedding from a space with one dimension per word to a continuous vector space with much lower dimension. Methods to generate this mapping include neural networks [17], dimensionality reduction on the word co-occurrence matrix [13–15], probabilistic models [9], and explicit representation in terms of the context in which words appear. For each piece of a text such as answer text and question body we constructed the centroid vector from the vectors of all words in that text.

$$centroid(w_1..w_n) = \frac{\sum_{i=1}^{n} w_i}{n} \tag{1}$$

We used various feature similarities calculated using the centroid word vectors on the question body and on the answer text, as well as on parts thereof:

Question to answer similarity. We assume that a good answer should have a centroid vector that is close to that for the question. We choose the question body to answer text vector similarities.

Dependency syntax tree based word vector similarity. We obtained the dependency syntax tree with the Stanford parser [4], and calculated similarities between centroid vectors of noun phrases from the answer text and the centroid vector of the noun phrases from the question body text. The assumption is that the same parts of dependency syntax tree between the question and the answer might be closer than other parts of dependency tree.

Aligned similarity. For each word in the question body, we chose the most similar word from the answer text and took the average of all the best word pair similarities as suggested in [25]

LDA topic similarity. We performed topic clustering using Latent Dirichlet Allocation (LDA) as implemented in the gensim toolkit [21]. We built topic models with 50 topics. The assumption here is that if the question and the answer share similar topics, they are more likely to be relevant to each other.

For the similarity measures mentioned above, we measure their cosine similarity as,

$$\frac{\vec{a} \cdot \vec{b}}{||\vec{a}|| \cdot ||\vec{b}||} \tag{2}$$

3.2 Metadata Features

Besides semantic features described above, we also used some metadata features:

Question length. If a question is longer, it may be more clear, which may help users to give a more relevant answer [6].

Answer length. The assumption here is that longer answers could bring more useful details [33].

Question to answer length. If a question is long and the answer is short, it may be less relevant.

3.3 Focusing Attention Network

In answer ranking, given a question $Q = \{q_1, q_2, q_3, ..., q_n\}$ where q_i is i-th word ,n is the question length, we can compute its representation in bidirectional LSTM architecture as follows:

$$X = D[q_1, q_2, ..., q_n] \tag{3}$$

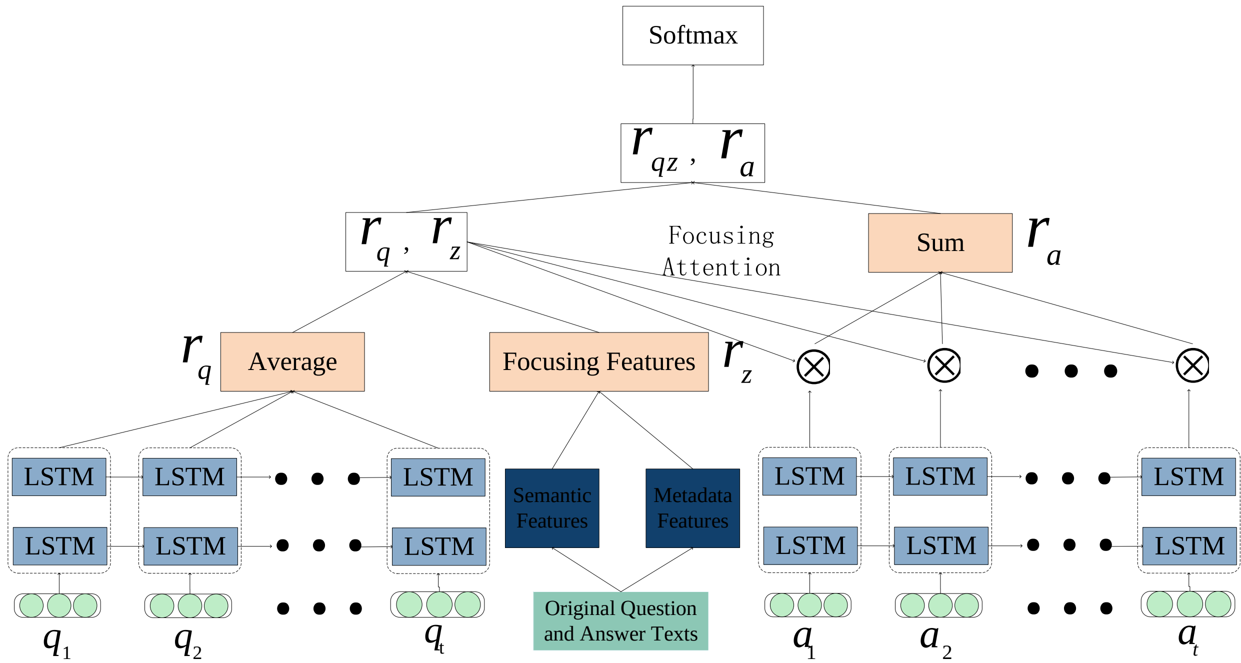


Figure 2: The architecture of our model.

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4}$$

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{5}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{6}$$

$$\tilde{c_t} = \tanh(W_c x_t + U_c h_{t-1} + b_c) \tag{7}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot \tilde{c_t} \tag{8}$$

$$h_t = o_t \odot \tanh(C_t) \tag{9}$$

where D is an embedding matrix that projects word to its embedding space in R^d; W, U are weight matrices and b are bias vectors; σ is the active function such as tanh.

In the LSTM architecture, there are three gates (input i, forget f and output o) and a cell memory vector c. The input gate can determine how incoming vectors x_t alter the state of the memory cell. The output gate can allow the memory cell to have an effort on the outputs. Finally, the forget gate allows the cell to remember or forget its previous state. Single direction LSTMs suffer a weakness of not utilizing the contextual information from the future tokens. Bidirectional LSTM utilizes both the previous and future context by processing the sequence on two directions, and generates two independent sequences of LSTM output vectors. One processes the input sequence in the forward direction, while the other process the input in the reverse direction. The output at each time step is the concatenation of the two output vectors from both directions,ie. $h_t = \overrightarrow{h_t} || \overleftarrow{h_t}$.

After the bidirectional LSTM, the last hidden variable h_n or all hidden states average $\frac{1}{n}\sum_{t=1}^{n} h_t$ is adopted as the preliminary question representation r_q. We can also obtain the last hidden variable H_a for the answer sentence, where $h_a(i)$ means the last hidden output of the i-th word in the answer.

$$H_a = [h_a(1), h_a(2), \ldots, h_a(m)] \tag{10}$$

When modeling the candidate answer sentence with length m: $S = s_1, s_2, s_3, \ldots, s_m$ in multiple features based attention model, instead of using the last hidden state or average hidden states, we use multiple features based attentive hidden states that are weighted by r_q and r_z, where r_z is the merge of semantic feature vector r_s and the metadata feature vector r_m:

$$s_t \propto f_{attention}(r_q, r_z, h_a(t)) \tag{11}$$

$$\tilde{h_a}(t) = h_a(t) s_t \tag{12}$$

$$r_a = \sum_{t=1}^{m} \tilde{h_a}(t) \tag{13}$$

where $h_a(t)$ is hidden state of the answer at time t. The attention function $f_{attention}$ is computed as:

$$m(t) = \tanh(W_{hm} h_a(t) + W_{qm} r_q + W_{sm} r_z) \tag{14}$$

$$f_{attention}(r_q, r_z, h_a(t)) = \exp(w_{ms}^T m(t)) \tag{15}$$

W_{hm}, W_{qm} and W_{sm} are attentive weight matrices and w_{ms} is attentive weight vector. So we can expect that the candidate answer sentence representation r_a may be represented in a question and different kinds of features guided way: when its hidden state $h_a(t)$ is irrelevant to the question and the different kinds of features(determined by attention weight s_t), it will take less part in the final representation; but when this hidden state is relavent to the question and different kinds of features, it will contribute more in representing r_a.

Table 1: Experimental dataset on the SemEval-2016 Task 3

	train	dev	test
number of questions	2,669	244	327
number of answers	17,900	2,440	3,270

Table 2: Experimental dataset on the SemEval-2017 Task 3

	train	dev	test
number of questions	4,879	244	293
number of answers	36,198	2,440	2,930

The final question vector r_{qz} and the answer vector r_a which we obtain using multiple features based attention are both useful in the ranking task. We splice them by the row and get the high level representation of the sentence $\overline{s}$. We use it to classify the answer of the question by a softmax function

$$p = softmax(W_c\overline{s} + b_c) \tag{16}$$

$$L = -\sum_{s} \log p_{sj} \tag{17}$$

where j is the label of answer s, W_c is the weight matrices and b_c is the bias vector.

4 EXPERIMENTS EVALUATION AND ANALYSIS

In this section, we describe our experimental setting. We first introduce the dataset and the competitive methods, then present the results and analysis.

4.1 Datasets Setup

We experiment with the dataset from SemEval-2016 Task3 and SemEval-2017 Task3 on answer ranking for Community Question Answering [18, 19]. The datasets contain question-answer threads from the Qatar Living forum. Each answer in the dataset is annotated with one of the following labels, reflecting how well it answers the question: Good, PotentiallyUseful, and Bad. At SemEval-2016 and SemEval-2017 Task3, the latter two classes were merged into Bad at testing time. The statistics of the datasets are summarized in Table 1 and Table 2.

4.2 Compared with Competitive Methods

We experimentally compare our model with the other models at SemEval-2016 and SemEval-2017 Task 3. They are Kelp, ConvKN, SemanticZ in SemEval-2016 and KeLP, Beihang-MSRA and ECNU in SemEval-2017.

- KeLP: Their system learned semantic relations between questions and answers using kernels and previously-proposed features from [3]. The System is based on the KeLP machine learning platform [7].
- ConvKN: They combined convolutional tree kernels and convolutional neural networks, together with text similarity and thread-specific features [10].

Table 3: Experimental Results on the SemEval-2016 Task 3

Model	MAP	AvgRec	MRR
Random baseline [1]	52.80	66.52	58.71
Search engine [1]	59.53	72.60	67.83
KeLP [1]	79.19	88.82	86.42
ConvKN [1]	77.66	88.05	84.93
SemanticZ [1]	77.58	88.14	85.21
Attention-BiLSTMs	77.60	88.27	85.05
FAN(only semantic features)	78.10	88.30	85.65
FAN(only metadata features)	78.20	88.38	85.73
FAN(both features)	**79.38**	**88.96**	86.18

- SemanticZ: They used the semantic similarity based on word embeddings and topics [16].
- KeLP: They used tree kernels with relational links in Support Vector Machine. The questions are aligned with the answers (or with the other questions) by means of a special REL tag, directly annotated in the parse trees. It used the homonymous tree kernel software toolkit, KeLP [8].
- Beihang-MSRA: They developed a ranking system for capturing semantic relations between text pairs with little word overlap. In addition to traditional NLP features, they introduced several neural network based matching features which enable their system to measure text similarity beyond lexicons [32].
- ECNU: They explored a traditional machine learning method which used multiple types of features, e.g., Word Match Features, Topic Model-based Features, and Lexical Semantic Similarity Features. Additionally, they also built a Convolutional Neural Network (CNN) model to learn joint representation for question-comment (Q-C) pair. They combined two different methods to represent question-comment pair [29].

4.3 Experimental Settings

The questions and answers are tokenized and lemmatized using NLTK [7]. We removed the stop words and changed all the letters to lowercase. We used semantic word embeddings obtained from Word2Vec [2] models trained on different unannotated data sources including the QatarLiving and DohaNews [16]. The dimension of word embedding was 200. Tokens that did not appear in the pre-trained word embeddings were replaced with a special token, of which the embedding was initialized randomly. The number of nodes in the hidden layer of the bidirectional LSTM is 128. Therefore, the question vector r_{qz} and the answer vector r_a are 256 dimensions. We merge the features by stitching them, so we obtain the 512 dimensions feature as input. The activation we used in the bidirectional LSTM is Relu and the dropout rate is set as 0.5. To evaluate the model performance, we adopt the mean average precision(MAP), AvgRec and MRR which are the primary metrics used in QA [26, 27].

4.4 Results and Analysis

The experimental results on the SemEval-2016 and SemEval-2017 Task 3 are shown in Table 3 and Table 4.

Table 4: Experimental Results on the SemEval-2017 Task 3

Model	MAP	AvgRec	MRR
Random baseline [18]	62.30	70.56	68.74
Search engine [18]	72.61	79.32	82.37
KeLP [18]	88.43	93.79	92.82
Beihang-MSRA [18]	88.24	93.87	92.34
ECNU [18]	86.72	92.62	91.45
Attention-BiLSTMs	83.48	89.34	88.25
FAN(only semantic features)	85.10	90.56	89.85
FAN(only metadata features)	86.23	91.28	90.47
FAN(both features)	**88.72**	**93.92**	92.63

Table 5: Effectiveness of FAN Model

Question	Traditional attentive word(top 5)	FAN word(top 5)	Answer
is there any place i can find scented massage oils in qatar?	massage, oils, qatar, affordable, prices	naseem, alnadir, smartlink, chinese, prices	It's called Naseem Alnadir. Right next to the Smartlink shop. You'll find the chinese salesgirls at affordable prices there.

We can see that our MAP(with both features) respectively improvement of 19.85 and 16.11 compared with the results obtained by the search engine in SemEval-2016 and SemEval-2017. We also compare the FPN(only semantic features), the FPN(only metadata features) and the FPN(both features) models and find that both of the features are effective to solve the attention divergence problem, especially the metadata features. The model with both features obtain the best result. KeLP used various kinds of features and got the best results in SemEval-2016 and SemEval-2017 which demonstrates that feature engineering is effective in answer ranking task. ConvKN ranked second in SemEval-2016. Beihang-MSRA obtained the best AvgRec and ECNU ranked fourth in SemEval-2017. Our MAP separately improves 0.19 and 0.29 compared with the results obtained by the KeLP in SemEval-2016 and SemEval-2017. Our model also gets the best AvgRec above the other methods in SemEval-2016 and SemEval-2017. The result proves that our model can outperform the state-of-the-art systems.

We also compare our model with the traditional attention-biLSTMs model. The result shows that our MAP respectively improves 1.78 and 5.24 in SemEval-2016 and SemEval-2017. Our **FAN** model can focus attention back on the right regions of the target terms and achieve significant improvements over the other models.

4.5 Case Study

In order to better understand our proposed model, we provided an example to show why our FAN model is more effective for the answer ranking task. As shown in Table 5, for the question "*is there any place i can find scented massage oils in qatar?*", it is required to find a place, and we observe that in the traditional attention model, the top 2 attentive words associate with the question are not location names, such as *massage* and *oils*. However, in the FAN model, we observe that top 2 attentive words associate with the question are location names, such as *naseem* and *alnadir*. In particular, these two words contain the information that associate with the key word "*Naseem Alnadir*" in the good answer, which solves the issues of attention divergence for answer ranking task. Therefore, our FAN model, which enhances the question representations to rank good answers ahead of bad ones.

5 CONCLUSION

In this work, we put forward the *attention divergence* to explain the poor performance of existing attention based methods of answer ranking on complicated long question-answer pairs, and propose a novel FAN to solve this problem. Different from the existing methods, firstly, we obtain the question representation with Bidirectional LSTMs, next we extract semantic features, metadata features from questions and their answers to form the focusing features, then we use the focusing attentive network to obtain answer representation. At last, we feed the question and answer representation to the softmax classifier for classification. Experiments on SemEval-2016 and SemEval-2017 have shown the effectiveness of our proposed model, especially in terms of the Mean Average Precision and precision.

Because semantic and metadata features are the important features in the sentence and the complicated long sentences can be seen in many tasks such as text classification and reading comprehension. In future work, we plan to extend the proposed idea to text classification and other related tasks.

REFERENCES

[1] PreslavNakov LluısMarquez AlessandroMoschitti, WalidMagdy HamdyMubarak AbedAlhakimFreihat, James Glass, and Bilal Randeree. 2016. SemEval-2016 Task 3: Community Question Answering. *Proceedings of SemEval* (2016), 525–545.

[2] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[3] Alberto Barrón-Cedeno, Simone Filice, Giovanni Da San Martino, Shafiq R Joty, Lluís Màrquez, Preslav Nakov, and Alessandro Moschitti. 2015. Thread-Level Information for Comment Classification in Community Question Answering.. In *ACL (2)*. Citeseer, 687–693.

[4] Marie-Catherine De Marneffe and Christopher D Manning. 2008. The Stanford typed dependencies representation. In *Coling 2008: Proceedings of the workshop on Cross-Framework and Cross-Domain Parser Evaluation*. Association for Computational Linguistics, 1–8.

[5] Wenzheng Feng, Yu Wu, Wei Wu, Zhoujun Li, and Ming Zhou. [n. d.]. Beihang-MSRA at SemEval-2017 Task 3: A Ranking System with Neural Matching Features for Community Question Answering. ([n. d.]).

[6] Alejandro Figueroa. 2017. Automatically generating effective search queries directly from community question-answering questions for finding related questions. *Expert Systems with Applications* 77 (2017), 11–19.

[7] Simone Filice, Danilo Croce, Alessandro Moschitti, and Roberto Basili. 2016. KeLP at SemEval-2016 Task 3: Learning semantic relations between questions and answers. *Proceedings of SemEval* 16 (2016), 1116–1123.

[8] Simone Filice, Giovanni Da San Martino, and Alessandro Moschitti. 2017. KeLP at SemEval-2017 task 3: Learning pairwise patterns in community question answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation. Vancouver, Canada, SemEval*, Vol. 17. 327–334.

[9] Amir Globerson, Gal Chechik, Fernando Pereira, and Naftali Tishby. 2007. Euclidean embedding of co-occurrence data. *Journal of Machine Learning Research* 8, Oct (2007), 2265–2295.

[10] Shafiq Joty, Alessandro Moschitti, Fahad A Al Obaidli, Salvatore Romeo, Kateryna Tymoshenko, and Antonio Uva. 2016. ConvKN at SemEval-2016 Task 3: Answer and question selection for question answering on Arabic and English fora. *Proceedings of SemEval* (2016), 896–903.

[11] Suyoun Kim, Takaaki Hori, and Shinji Watanabe. 2017. Joint CTC-attention based end-to-end speech recognition using multi-task learning. In *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*. IEEE, 4835–4839.
[12] Yuta Koreeda, Takuya Hashito, Yoshiki Niwa, Misa Sato, Toshihiko Yanase, Kenzo Kurotsuchi, and Kohsuke Yanai. 2017. bunji at SemEval-2017 task 3: Combination of neural similarity features and comment plausibility features. In *Proceedings of the International Workshop on Semantic Evaluation. Vancouver, Canada, SemEval*, Vol. 17. 353–359.
[13] Rémi Lebret and Ronan Collobert. 2013. Word emdeddings through hellinger PCA. *arXiv preprint arXiv:1312.5542* (2013).
[14] Omer Levy and Yoav Goldberg. 2014. Neural word embedding as implicit matrix factorization. In *Advances in neural information processing systems*. 2177–2185.
[15] Yitan Li, Linli Xu, Fei Tian, Liang Jiang, Xiaowei Zhong, and Enhong Chen. 2015. Word Embedding Revisited: A New Representation Learning and Explicit Matrix Factorization Perspective.. In *IJCAI*. 3650–3656.
[16] Todor Mihaylov and Preslav Nakov. 2016. SemanticZ at SemEval-2016 Task 3: Ranking relevant answers in community question answering using semantic similarity based on fine-tuned word embeddings. *Proceedings of SemEval* (2016), 879–886.
[17] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems*. 3111–3119.
[18] Preslav Nakov, Doris Hoogeveen, Lluís Màrquez, Alessandro Moschitti, Hamdy Mubarak, Timothy Baldwin, and Karin Verspoor. 2017. SemEval-2017 Task 3: Community Question Answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval '17)*. Association for Computational Linguistics, Vancouver, Canada.
[19] Preslav Nakov, Lluís Màrquez, Alessandro Moschitti, Walid Magdy, Hamdy Mubarak, Abed Alhakim Freihat, Jim Glass, and Bilal Randeree. 2016. SemEval-2016 Task 3: Community Question Answering. In *Proceedings of the 10th International Workshop on Semantic Evaluation (SemEval '16)*. Association for Computational Linguistics, San Diego, California.
[20] Xipeng Qiu and Xuanjing Huang. 2015. Convolutional Neural Tensor Network Architecture for Community-Based Question Answering.. In *IJCAI*. 1305–1311.
[21] Radim Rehurek and Petr Sojka. 2010. Software framework for topic modelling with large corpora. In *In Proceedings of the LREC 2010 Workshop on New Challenges for NLP Frameworks*. Citeseer.
[22] Aliaksei Severyn and Alessandro Moschitti. 2013. Automatic feature engineering for answer selection and extraction. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*. 458–467.
[23] Gen Takahashi, Takeshi Yamada, Shoji Makino, and Nobutaka Ono. 2016. Acoustic scene classification using deep neural network and frame-concatenated acoustic feature. *Detection and Classification of Acoustic Scenes and Events* (2016).
[24] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. Lstm-based deep learning models for non-factoid answer selection. *arXiv preprint arXiv:1511.04108* (2015).
[25] Quan Hung Tran, Vu Tran, Tu Vu, Minh Le Nguyen, and Son Bao Pham. 2015. JAIST: Combining multiple features for answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval*, Vol. 15. 215–219.
[26] Bingning Wang, Kang Liu, and Jun Zhao. 2016. Inner Attention based Recurrent Neural Networks for Answer Selection.. In *ACL (1)*.
[27] Di Wang and Eric Nyberg. 2015. A long short-term memory model for answer sentence selection in question answering. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, Vol. 2. 707–712.
[28] Mengqiu Wang and Christopher D Manning. 2010. Probabilistic tree-edit models with structured latent variables for textual entailment and question answering. In *Proceedings of the 23rd International Conference on Computational Linguistics*. Association for Computational Linguistics, 1164–1172.
[29] Guoshun Wu. [n. d.]. Yixuan Sheng adn Man Lan, and Yuanbin Wu. 2017a. ECNU at SemEval-2017 task 3: Using traditional and deep learning methods to address community question answering task. In *Proceedings of the Workshop on Semantic Evaluation. Vancouver, Canada, SemEval*, Vol. 17. 365–369.
[30] Guoshun Wu and Man Lan. 2016. ECNU at SemEval-2016 Task 3: Exploring traditional method and deep learning method for question retrieval and answer ranking in community question answering. *Proceedings of SemEval* (2016), 872–878.
[31] Guoshun Wu, Yixuan Sheng, Man Lan, and Yuanbin Wu. 2017. ECNU at SemEval-2017 Task 3: Using Traditional and Deep Learning Methods to Address Community Question Answering Task. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*. 365–369.
[32] Yu Wu, WenZheng Feng, Wei Wu, Ming Zhou, and Zhoujun Li. 2017. Beihang-MSRA at SemEval-2017 task 3: A ranking system with neural matching features for Community Question Answering. In *Proceedings of the 11th International Workshop on Semantic Evaluation. Vancouver, Canada, SemEval*, Vol. 17. 281–287.
[33] Shuo Yang, Lei Zou, Zhongyuan Wang, Jun Yan, and Ji-Rong Wen. 2017. Efficiently Answering Technical Questionsï£¡ï£¡A Knowledge Graph Approach. *Thirty-First AAAI Conference on Artificial Intelligence* (2017).
[34] Scott Wen-tau Yih, Ming-Wei Chang, Chris Meek, and Andrzej Pastusiak. 2013. Question answering using enhanced lexical semantic models. (2013).
[35] Naiyun Zhou, Andrey Fedorov, Fiona Fennessy, Ron Kikinis, and Yi Gao. 2017. Large scale digital prostate pathology image analysis combining feature extraction and deep neural network. *arXiv preprint arXiv:1705.02678* (2017).