Enhancing Recurrent Neural Networks with Positional Attention for Question Answering

Qin Chen[1], Qinmin Hu[1], Jimmy Xiangji Huang[2], Liang He[1,3] and Weijie An[1]
[1]Department of Computer Science & Technology, East China Normal University, Shanghai, China
[2]Information Retrieval & Knowledge Management Research Lab, York University, Toronto, Canada
[3]Shanghai Engineering Research Center of Intelligent Service Robot, Shanghai, China
{qchen,wjan}@ica.stc.sh.cn,{qmhu,lhe}@cs.ecnu.edu.cn,jhuang@yorku.ca

ABSTRACT

Attention based recurrent neural networks (RNN) have shown a great success for question answering (QA) in recent years. Although significant improvements have been achieved over the non-attentive models, the position information is not well studied within the attention-based framework. Motivated by the effectiveness of using the word positional context to enhance information retrieval, we assume that if a word in the question (i.e., *question word*) occurs in an answer sentence, the neighboring words should be given more attention since they intuitively contain more valuable information for question answering than those far away. Based on this assumption, we propose a positional attention based RNN model, which incorporates the positional context of the question words into the answers' attentive representations. Experiments on two benchmark datasets show the great advantages of our proposed model. Specifically, we achieve a maximum improvement of 8.83% over the classical attention based RNN model in terms of mean average precision. Furthermore, our model is comparable to if not better than the state-of-the-art approaches for question answering.

基于的假设是，如果question中的某个词出现在answer中，那answer中这个词附近的词应该得到更多的关注。

1 INTRODUCTION

Recurrent neural networks (RNN) have been widely used for question answering (QA) due to its good performance [8–10]. In the RNN based QA models, each word in a question or an answer sentence is represented with a hidden vector first. Then, all the hidden vectors are aggregated for sentence representations. Afterwards, the best answer is selected from a candidate answer pool according to the sentence similarity.

One major challenge in RNN is how to aggregate the hidden vectors for sentence representations. Recently, the attention mechanism has shown its effectiveness for the attentive sentence representations in many NLP tasks including QA [1, 8, 9, 14]. In particular, a weight is automatically generated for each word via attention, and the sentence is represented as the weighted sum of the hidden vectors. Various attention mechanisms have been proposed in previous studies. In [8], the attentive weights of the words in answers relied on the hidden representation of questions. Santos et al. [1] proposed a two-way attention mechanism, where the attentive weights for the question (answer) were influenced by the answer (question) representation according to the word-by-word interaction matrix. However, these attentions relied on the hidden vectors, which may excessively concern the words near the end of the sentence due to the abundant semantic accumulation over the word sequence in RNN. To alleviate the attention bias problem, Wang et al. [9] proposed three inner attention methods, which added the attention information before the hidden representations and achieved the state-of-the-art performance in QA.

To the best of our knowledge, all the previous attention mechanisms neglect the positional context, which has been extensively studied for performance boosting in information retrieval (IR). In [3] and [17], the occurrence positions of the query terms were modeled with various kernels and then integrated into traditional IR models to enhance the retrieval performance. Inspired by the effectiveness of the positional context in IR, we attempt to incorporate it into classical attentions to enhance the performance of RNN based QA. Specifically, it is assumed that if a word in the question (i.e., *question word*) occurs in an answer sentence, it will have an influence on the neighboring context. In other words, the neighboring words should be given more attention than those far away since they may contain more question relevant information. Based on this assumption, we propose a **Po**sitional **A**ttention based **RNN** (**RNN-POA**) model, which models the position-aware influence of the question word for answers' attentive representations. To be specific, we first present a position-aware influence propagation strategy, in which the influence of the question word is propagated to other positions in the answer sentence by a distance-sensitive kernel. Then, a position-aware influence vector for each word is generated in the hidden space, according to the accumulated influence propagated by all the question words occurring in the answer. After that, the position-aware influence vector is integrated into the classical attention mechanism for answers' attentive representations. We perform experiments on two publicly available benchmark datasets, namely TREC-QA and WikiQA. The results show that our positional attention can significantly outperform the classical attention which does not involve any position information. Furthermore, the performance of our proposed RNN-POA model is comparable to the state-of-the-art approaches for question answering.

The main contributions of our work are as follows: (1) as far as we know, it is the first attempt to investigate the effectiveness of the positional context for answers' attentive representations; (2) we propose a positional attention based RNN model, which has been proved to be effective to boost the QA performance; (3) our positional attention approach can help alleviate the attention bias

SIGIR '17, August 07-11, 2017, Shinjuku, Tokyo, Japan

ACM ISBN 978-1-4503-5022-8/17/08…$15.00
https://doi.org/10.1145/3077136.3080699

problem by utilizing the position information of the question words, instead of the semantic accumulated hidden representations.

2 PROPOSED MODEL

2.1 Framework of RNN-POA

Figure 1(b) shows the framework of our positional attention based RNN (RNN-POA) model for answer representations. To have a better comparison, the classical attention based question representation is also shown in Figure 1(a). We adopt the bidirectional long short-term memory (BLSTM) [2] model for sentence modeling, which takes the pre-trained word embeddings as the input, and generates the hidden vectors by recurrent updates [2].

普通的self-attention

$$u_i = \tanh(W_s h_i + b_s),$$
$$\alpha_i = \frac{\exp(u_i^\top u_s)}{\sum_i \exp(u_i^\top u_s)},$$
$$v = \sum_i \alpha_i h_i,$$

To obtain the composite representations of the questions, we adopt the classical attention used in [14], which solely relies on the hidden vectors for the attentive weight generation. Regarding to the answer, we propose a positional attention approach, and perform additional steps upon the classical attention as follows: (1) find the occurrence positions of the question words in an answer sentence; (2) propagate the influence of the question words to other positions with our position-aware influence propagation strategy; (3) generate the position-aware influence vector for each word in the answer sentence according to the propagated influence; (4) incorporate the position-aware influence vector into the classical attention mechanism.

With the attentive representations of both questions and answers, various similarity functions can be utilized to measure the relevance between them. We utilize the Manhattan distance similarity function with l_1 norm (Formula (1)), which performs slightly better than the other alternatives such as cosine similarity as indicated in [5]:

$$sim(r_q, r_a) = exp(-||r_q - r_a||_1) \tag{1}$$

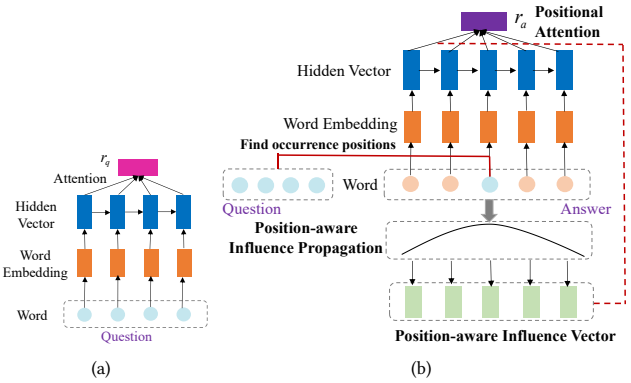


Figure 1: (a) question representation with classical attention based RNN; (b) answer representation with positional attention based RNN.

2.2 Position-aware Influence Propagation

Based on our previous assumption, a question word will have an influence on the neighboring context if it occurs in an answer sentence. Here we model the position-aware influence propagation with the Gaussian kernel, which has been proved to be effective for position modeling in IR [3, 4, 17]:

$$Kernel(u) = exp(\frac{-u^2}{2\sigma^2}) \tag{2}$$

σ 的最优值会根据不同的词而不同，但本文使用一个常数。

where u is the distance between the question word and the current word, σ is a parameter which restricts the propagation scope, and $Kernel(u)$ denotes the obtained influence corresponding to the distance of u based on the kernel.

Note that the position-aware influence is diminishing when the distance increases. In particular, when $u = 0$ (i.e., the current word is exactly a question word), the maximum propagated influence is obtained. As to the propagation scope σ, the optimal value may vary from words to words. In this paper, we apply a constant σ value for all question words, and focus on incorporating the positional context into attentions.

2.3 Position-aware Influence Vector

In this section, in order to model the influence in a high-dimensional space for attentions, we will demonstrate how to obtain the position-aware influence vector for each word in an answer sentence. First, we assume that the influence for a specific distance follows the Gaussian distributions over the hidden dimensions. Then, an influence base matrix K is defined based on the assumption, where each column denotes the influence base vector corresponding to a specific distance. More formally, each element of K is defined as:

$$\mathbf{K}(i, u) \sim N(Kernel(u), \sigma') \tag{3}$$

where $\mathbf{K}(i, u)$ denotes the influence corresponding to the distance of u in the i's dimension, and N is the normal density with an expected value of $Kernel(u)$ and standard deviation of σ'.

With the influence base matrix, the influence vector for a word at a specific position is obtained by accumulating the influence of all the question words occurring in the answer:

$$\mathbf{p}_j = \mathbf{K}\mathbf{c}_j \tag{4}$$

where $\mathbf{p}_j$ denotes the accumulated influence vector for the word at position j, and $\mathbf{c}_j$ is a distance count vector which measures the count of question words with various distances. Specifically, for the word at position j, the count of question words with a distance of u, namely $c_j(u)$, is calculated as:

$$c_j(u) = \sum_{q \in Q} [(j-u) \in pos(q)] + [(j+u) \in pos(q)] \tag{5}$$

where Q represents a question containing multiple question words, q is a word in Q, $pos(q)$ denotes the set of q's occurrence positions in an answer sentence, and $[\cdot]$ is an indicator function which equals to 1 if the condition satisfies and otherwise equals to 0.

2.4 Positional Attention

In most previous attention mechanisms, the attentive weight of a word relies on the hidden representations, while the position information is not well investigated. In this section, we propose a positional attention approach, which incorporates the position-aware influence of the question words into answers' attentive representations. Specifically, the attentive weight of a word at position j in the answer sentence is formulated as:

$$\alpha_j = \frac{exp(e(\mathbf{h}_j, \mathbf{p}_j))}{\sum_{k=1}^{l} exp(e(\mathbf{h}_k, \mathbf{p}_k))} \tag{6}$$

where $\mathbf{h}_j$ is the hidden vector at position j based on RNN, $\mathbf{p}_j$ is the accumulated position-aware influence vector obtained by Formula (4), l denotes the sentence length, and $e(\cdot)$ is a score function which measures the word importance based on the hidden vector and the position-aware influence vector. More formally, the score function is defined as:

$$e(\mathbf{h}_j, \mathbf{p}_j) = \mathbf{v}^T tanh(\mathbf{W}_H \mathbf{h}_j + \mathbf{W}_P \mathbf{p}_j + \mathbf{b}) \quad (7)$$

where $\mathbf{W}_H$ and $\mathbf{W}_P$ are matrices, $\mathbf{b}$ is a bias vector, $tanh$ is the hyperbolic tangent function, $\mathbf{v}$ is a global vector and $\mathbf{v}^T$ denotes its transpose. $\mathbf{W}_H$, $\mathbf{W}_P$, $\mathbf{b}$ and $\mathbf{v}$ are the parameters.

With the obtained attentive weights, an answer sentence is represented by the weighted sum of all the hidden vectors:

$$r_a = \sum_{j=1}^{l} \alpha_j \mathbf{h}_j \quad (8)$$

3 EXPERIMENTS

3.1 Experimental Setup

Datasets and Evaluation Metrics. We conduct experiments on two public question answering datasets: TREC-QA and WikiQA. TREC-QA was created by Wang et al. [11] based on the TREC QA track data. WikiQA [13] is an open domain question-answering dataset in which all answers are collected from the Wikipedia. Each dataset is split into 3 parts, i.e., train, dev and test, and the statistics are presented in Table 1. To evaluate the model performance, we adopt the mean average precision (MAP) and mean reciprocal rank (MRR), which are the primary metrics used in QA [1, 9].

Table 1: Dataset Statistics. "Avg QL" and "Avg AL" denote the average length of questions and answers.

Dataset	# of quetions (train/dev/test)	Avg QL (train/dev/test)	Avg AL (train/dev/test)
TREC-QA	1162/65/68	7.57/8.00/8.63	23.21/24.9/25.61
WikiQA	873/126/243	7.16/7.23/7.26	25.29/24.59/24.59

Parameter Settings. For the word embeddings, we use the 100-dimensional GloVe [6] word vectors[1]. The parameters in BLSTM are shared between questions and answers, which has been shown to be effective to improve the performance [9]. The dimension of the hidden vectors and the position-aware influence vectors is set to 50. Regarding to the propagation scope σ (in Formula (2)), we investigate a list of values ranging from 5 to 55 with an interval of 10. The value of σ' (in Formula (3)) is empirically set to 0.1. We adopt the cross-entropy loss as the training objective, and utilize the Adadelta [16] algorithm for parameter update. The optimal parameters are obtained based on the best MAP performance on the development (dev) set.

3.2 Effect of Positional Attention

To investigate the effect of our positional attention approach, two basic baselines which do not involve the position information, namely average pooling ("**AVG**") [10] and the classical attention ("**ATT**") used in [14], are integrated into the BLSTM based RNN model for comparisons. Table 2 shows the performance of various models. Statistical significant tests are performed based on the paired t-test at the 0.05 level. The symbols as $*$ and $\triangle$ represent significant improvements over "AVG" and "ATT" respectively. We observe that the classical attention mechanism slightly outperforms the average pooling method by capturing part of the informative words in answers. However, it does not pay specifical attention to the question words and their surrounding context, which loses some useful information for question answering. In our positional attention approach, the importance of the question word and the surrounding context is explicitly highlighted via the position-aware influence propagation of the question words. Therefore, we can achieve significant improvements over the two baselines on both QA datasets, and the maximum improvement is as high as 8.83% in terms of MAP.

Table 2: Performance of various models.

Model	TREC-QA		WikiQA	
	MAP	MRR	MAP	MRR
RNN-AVG	0.7064	0.8086	0.6889	0.6999
RNN-ATT	0.7180	0.8121	0.6961	0.7085
RNN-POA	$\mathbf{0.7814}^{*}_{\triangle}$	$\mathbf{0.8513}^{*}_{\triangle}$	$\mathbf{0.7212}^{*}_{\triangle}$	$\mathbf{0.7312}^{*}_{\triangle}$

3.3 Performance Comparisons

To further evaluate the effectiveness of our proposed model, we compare it with the recent work in QA. Table 3 and Table 4 summarize the results on TREC-QA and WikiQA respectively. For TREC-QA, five strong baselines are used for comparisons: (1) a combination of the BLSTM model and BM25 model [10]; (2) inner attention based RNN models which added the attention information before the hidden representations [9]; (3) a convolutional neural network (CNN) based architecture using both the hidden features and the statistical features for ranking [7]; (4) a learning-to-rank method which leveraged the word alignment features and lexical features for ranking [12]; (5) an extended LSTM framework which incorporated CNN and built the attention matrix after sentence representations [8]. As to WikiQA, in addition to [8] and [9] mentioned above, we make a comparison with other two strong baselines: (1) a bigram CNN model with average pooling [13]; (2) a CNN model which used an interactive attention matrix for the attentive representations [15].

It is observed that we achieve the new state-of-the-art performance on TREC-QA in terms of both MAP and MRR. Regarding to WikiQA, our RNN-POA model outperforms all the strong baselines except the inner attention based RNN models [9]. This validates our previous assumption that the words close to the question words should be given more attention than those far away. In addition, it has been proved to be effective for incorporating the positional context into answers' attentive representations.

3.4 Investigation of Propagation Scope σ

The parameter σ (in Formula (2)) controls the influence propagation scope of the question words. For a certain distance, the propagated position-aware influence increases when σ grows. Figure 2 plots the MAP and MRR metrics over a set of σ values ranging from 5 to 55 with a step of 10. We observe that the general tendency for each

[1] http://nlp.stanford.edu/data/glove.6B.zip

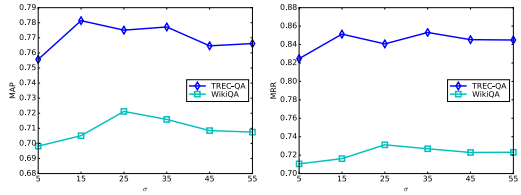
Table 3: Performance comparisons on TREC-QA. The work marked with † used the cleaned dataset.

System	MAP	MRR
Wang and Nyberg (2015) [10]	0.7134	0.7913
Wang et al. (2016) [9] †	0.7369	0.8208
Severyn and Moschitti (2015) [7]	0.7459	0.8078
Wang and Ittycheriah (2015) [12]	0.7460	0.8200
Santos et al. (2016) [8]	0.7530	0.8511
RNN-POA	**0.7814**	**0.8513**

Table 4: Performance comparisons on WikiQA

	System	MAP	MRR
	Yang et al. (2015) [13]	0.6520	0.6652
	Santos et al. (2016) [8]	0.6886	0.6957
	Yin et al. (2015) [15]	0.6921	0.7108
IARNN	Wang et al. (2016) [9]	**0.7341**	**0.7418**
	RNN-POA	0.7212	0.7312

evaluation metric is similar. Specifically, the performance increases when σ grows at first. Then, it decreases and tends to be stable when σ becomes larger. On the whole, a σ value between 15 and 35 is recommended to be a reliable setting in our experiments.



(a) MAP (b) MRR

Figure 2: Impact of the propagation scope σ

3.5 A Case Study

To have an intuitive understanding of our proposed RNN-POA model, we draw a word heatmap for a case based on the classical attention and our positional attention respectively in Figure 3. Obviously, to answer this question, we should focus on the words "*George Warrington*". However, the classical attention cares more about some irrelevant words such as "*market*". Although these words have some semantic relations with the words in the question (e.g., the word "*market*" co-occurs frequently with the words "*chief*" and "*executive*"), the semantically related words are not necessarily useful for question answering. In contrast, our positional attention approach concerns more about the question words such as "*Amtrak*", "*president*", "*chief*" and "*executive*", as well as the surrounding context such as "*George Warrington*", which provides more valuable clues for question answering. That is why our proposed positional attention approach can achieve much better performance than the classical attention method.

4 CONCLUSIONS

In this paper, we propose a positional attention based RNN (RNN-POA) model, which incorporates the positional context of the question words into answers' attentive representations. The experimental results on two benchmark datasets show the overwhelming

Q: who is the president or chief executive of amtrak?

RNN-ATT "Long term success here has to do with doing it right , getting it right and increasing market share." said George Warrington, Amtrak's president and chief executive.

RNN-POA "Long term success here has to do with doing it right , getting it right and increasing market share." said George Warrington, Amtrak's president and chief executive.

Figure 3: An example of RNN-ATT and RNN-POA.

superiority of our proposed model over the basic baselines which do not incorporate any position information. Furthermore, compared with the state-of-the-art approaches in question answering, our proposed model can achieve a considerable performance by merely incorporating the position information into the classical attention mechanism. In the future, we will investigate more strategies to model the position influence of the question words.

ACKNOWLEDGMENTS

This research is supported by the National Key Technology Support Program (No.2015BAH01F02), Science and Technology Commission of Shanghai Municipality (No.16511102702), and Shanghai Municipal Commission of Economy and Information Under Grant Project (No.201602024). This research is also supported by a Discovery grant from the Natural Sciences and Engineering Research Council (NSERC) of Canada and an NSERC CREATE award.

REFERENCES

[1] Cıcero Nogueira dos Santos, Ming Tan, Bing Xiang, and Bowen Zhou. 2016. Attentive pooling networks. *arXiv preprint arXiv:1602.03609* (2016).
[2] Alex Graves and Jürgen Schmidhuber. 2005. Framewise phoneme classification with bidirectional LSTM and other neural network architectures. *Neural Networks* 18, 5 (2005), 602–610.
[3] Baiyan Liu, Xiangdong An, and Jimmy Xiangji Huang. 2015. Using term location information to enhance probabilistic information retrieval. In *SIGIR*. 883–886.
[4] Jun Miao, Jimmy Xiangji Huang, and Zheng Ye. 2012. Proximity-based rocchio's model for pseudo relevance. In *SIGIR*. 535–544.
[5] Jonas Mueller and Aditya Thyagarajan. 2016. Siamese Recurrent Architectures for Learning Sentence Similarity. In *AAAI*. 2786–2792.
[6] Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global Vectors for Word Representation. In *EMNLP*. 1532–1543.
[7] Aliaksei Severyn and Alessandro Moschitti. 2015. Learning to rank short text pairs with convolutional deep neural networks. In *SIGIR*. 373–382.
[8] Ming Tan, Cicero dos Santos, Bing Xiang, and Bowen Zhou. 2015. LSTM-based deep learning models for non-factoid answer selection. *arXiv:1511.04108* (2015).
[9] Bingning Wang, Kang Liu, and Jun Zhao. 2016. Inner attention based recurrent neural networks for answer selection. In *ACL*. 1288–1297.
[10] Di Wang and Eric Nyberg. 2015. A Long Short-Term Memory Model for Answer Sentence Selection in Question Answering.. In *ACL*. 707–712.
[11] Mengqiu Wang, Noah A Smith, and Teruko Mitamura. 2007. What is the Jeopardy Model? A Quasi-Synchronous Grammar for QA. In *EMNLP-CoNLL*. 22–32.
[12] Zhiguo Wang and Abraham Ittycheriah. 2015. FAQ-based Question Answering via Word Alignment. *arXiv preprint arXiv:1507.02628* (2015).
[13] Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. WikiQA: A Challenge Dataset for Open-Domain Question Answering.. In *EMNLP*. 2013–2018.
[14] Zichao Yang, Diyi Yang, Chris Dyer, Xiaodong He, Alex Smola, and Eduard Hovy. 2016. Hierarchical attention networks for document classification. In *NAACL-HLT*. 1480–1489.
[15] Wenpeng Yin, Hinrich Schütze, Bing Xiang, and Bowen Zhou. 2015. ABCNN: Attention-based convolutional neural network for modeling sentence pairs. In *arXiv preprint arXiv:1512.05193*.
[16] Matthew D Zeiler. 2012. ADADELTA: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701* (2012).
[17] Jiashu Zhao, Jimmy Xiangji Huang, and Ben He. 2011. CRTER: using cross terms to enhance probabilistic information retrieval. In *SIGIR*. 155–164.