Co-Attention and Aggregation Based Chinese Recognizing Textual Entailment Model[1]

Pengcheng Liu, Lingling Mu, Hongying Zan

College of information Engineering, Zhengzhou University, Zhengzhou, China
Liupengcheng2016@163.com,iellmu@zzu.edu.cn

Abstract. Recognizing Textual Entailment is a fundamental task of natural language processing, and its purpose is to recognize the inferential relationship between two sentences. With the development of deep learning and construction of relevant corpus, great progress has been made in English Textual Entailment. However, the progress in Chinese Textual Entailment is relatively rare because of the lack of large-scale annotated corpus. The Seventeenth China National Conference on Computational Linguistics (CCL 2018) first released a Chinese textual entailment dataset that including 100,000 sentence pairs, which provides support for application of deep learning model. Inspired by attention models on English, we proposed a Chinese recognizing textual entailment model based on co-attention and aggregation. This model uses co-attention to calculate the feature of relationship between two sentences, and aggregates this feature with another feature obtained from sentences. Our model achieved 93.5% accuracy on CCL2018 textual entailment dataset, which is higher than the first place in previous evaluations. Experimental results showed that recognition of contradiction relations is difficult, but our model outperforms other benchmark models. What's more, our model can be applied to Chinese document based question answer (DBQA). The accuracy of the experiment results on the dataset of NLPCC2016 is 72.3%.

Keywords: Textual Entailment, Co-attention, Aggregation, DBQA.

1 Introduction

Recognizing Textual Entailment (RTE), also known as natural language inference (NLI), is one of the important tasks in the field of natural language processing (NLP), and its achievement could be applied to other tasks such as Question Answer (QA), reading comprehension, etc. RTE is a study to determine whether there is a one-way semantic inferential relationship between two sentences. The two sentences are called as premise and hypothesis, respectively. According to whether the hypothesis can be

[1] The authors were supported financially by the National Social Science Fund of China (18ZDA315), Programs for Science and Technology Development in Henan province (No.192102210260) and the Key Scientific Research Program of Higher Education of Henan (No.20A520038).

inferred by the corresponding premise, the relationships between two sentences can be divided into three categories: entailment, contradiction and neural (Table 1).

Table 1. Textual Entailment samples

Premise	Hypothesis	Relationship
长颈鹿的嘴巴闭上了。 (The giraffe's mouth closed.)	长颈鹿不吃东西。 (The giraffe doesn't eat.)	Entailment
长颈鹿的嘴巴闭上了。 (The giraffe's mouth closed.)	长颈鹿的嘴巴张开。 (The giraffe's mouth is open.)	Contradiction
长颈鹿的嘴巴闭上了。 (The giraffe's mouth closed.)	长颈鹿的脖子长。 (The giraffe has a long neck.)	Neutral

In recent years, English textual entailment recognition has been developing rapidly because of the construction of large-scale annotated corpus [1] and the development of deep learning methods [2]. In deep learning neural models, attention based models perform well in English textual entailment recognition, which can effectively extract interactive information between two sentences.

In the field of Chinese recognizing textual entailment, Tan *et al.* [3] proposed the BiLSTM+CNN method and the accuracy achieved 61.9% in RITE2014 Chinese corpus. The hierarchical LSTM method proposed by Chen *et al.* [4] achieved 58.9% accuracy on reading comprehension data M2OCTE. However, the lack of Chinese large-scale corpus makes the use of deep learning model still relatively rare. Table 2 lists the common Chinese RTE datasets. The CCL2018 Chinese RTE task firstly releases a Chinese textual entailment corpus with 100,000 sentences pairs[2], which provide support for the use of deep neural methods, The current highest accuracy on this corpus is 82.38%. The attention-based models in Chinese textual entailment have a better condition for application due to the improvement of large-scale corpuses.

Table 2. Chinese Textual Entailment datasets

	train/test (number of sentences pairs)	Accuracy (%)
RITE2014	1,976/1,200	61.74[3]
M2OCTE	8,092/5,117	58.92[4]
CCL2018	90,000/10,000	82.38[2]

Inspired by the models of Decomp-Att [5] and SWEM [6], this paper proposes a textual entailment model based on co-attention and aggregation, which obtains 93.5% on the CCL2018 Chinese textual entailment recognition dataset. The accuracy exceeded the first place in previous evaluations of CCL2018. The model simultaneously calculates a weight matrix of two sentences through co-attention. The weight matrix that denote relationship between two sentences is interacted with encoding features through aggregation. In order to test the cross-task ability of the textual entailment model, this model was applied to QA tasks, and 72.3% accuracy was obtained on the

[2] http://www.cips-cl.org/static/CCL2018/call-evaluation.html#task3

DBQA dataset of NLPCC2016, exceeding the third place in the evaluation of NLPCC2016[3]. The contributions of this model are as follows:

(1) We successfully use co-attention in Chinese textual entailment to recognize the semantic inferential relationship between two sentences.

(2) We use aggregation to get more features of the encoding layer and reduce features loss.

(3) We add a pooling operation to enhance our model's ability of features extraction.

The rest of this paper included: Section 2 introduces the related backgrounds of the RTE models. Section 3 describes the model based on co-attention and aggregation in details. Section 4 shows the experimental results and discussions, and Section 5 is conclusion and prospect.

2 Related work

RTE is firstly proposed by Dagan *et al.* [7]. Early studies usually use evaluation datasets such as PASCAL [8], RTE [9] and SICK [10]. Then large-scale annotated corpuses such as SNLI and MultiNLI [11] are published, which facilitate the application of deep learning models on RTE task. The textual entailment methods based on deep learning have two types. The first method, uses the encoder to obtain the features of two sentences respectively, and then uses the aggregation to construct the corresponding feature. The second method uses attention to match words of two sentences, and obtains the feature vector of their relationship.

The first method is characterized by encoding sentences respectively. Bowman *et al.* [1] first use the neural network model to process textual entailment. They use LSTM and RNN to represent the sentence pairs in SNLI, and finally input the connection of these two representations into the multi-layer perceptron. Their experimental results on SNLI achieves 77.6% accuracy. Wang *et al.* [2] first use LSTM to encode and compare two sentences, then use attention to construct a weight matrix for them, and later use LSTM for matching. It is an early work of RTE combining attention and neural network. The SWEM model proposed by Shen *et al.* [6] is based on the max pooling and average pooling of word embedding. It is evaluated on 17 datasets including SNLI and MultiNLI, and most of them achieve the best accuracy. In addition, the InferSent model on the basis of BiLSTM proposed by Conneau *et al.* [12] reaches 84.5% on SNLI.

The second method is proposed based on the first method, which combines attention with sentences encoding and can obtain more interactive information. The attention mechanism that calculates sentences relationship becomes an important part of English RTE models. ESIM proposed by Chen *et al.* [13] consists of two parts: one uses the sequence model to collect the context information of words, and the other uses the tree model to collect the clause information. Both of them use attention to match words in sentences. It increases the accuracy of SNLI to above 88% for the

[3] http://tcci.ccf.org.cn/conference/2016/pages/page05_evadata.html

first time. Moreover, Decomp-Att proposed by Parikh *et al.* [5] is based on the matching mechanism of sentence pairs. It is characterized by combining the matching mechanism with the attention. It reaches 86.8% on SNLI with a simple structure. Furthermore, The MwAN model proposed by Tan *et al.* [14] combines four attention mechanisms. It achieves 89.4% on SNLI and 91.35% accuracy on sQuAD. The DR-BiLSTM proposed by Ghaeini et al. [15] demonstrates that enhancing the relationship between premise and hypothesis during coding helps to improve the model's effectiveness. Finally, the result of a DRCN model based on the co-attention and RNN proposed by Kim *et al.* [16] is the best on SNLI.

The above models have been widely used in the English textual entailment recognition, but are rarely applied to Chinese textual entailment. Inspired by the models of Decomp-Att, ESIM, SWEM and DR-BiLSTM, this paper proposed a Chinese textual entailment model that merges attention, pooling and aggregation. The model is characterized by combination of the co-attention mechanism and the aggregation mechanism. The structure of it is simpler than the traditional deep neural network models. The experiment results showed that the accuracy on the CCL2018 dataset is 93.5%, which is the best result on this dataset. Our model was also applied to the question and answer task, and achieved an accuracy of 72.3% on the NLPCC2016 document based QA dataset.

3 Model

Our model is called Co-Attention and Aggregation Model (CoAM). It's structure is shown in Figure 1. It consists of four parts: encoding layer, co-attention layer, aggregation layer and pooling layer. Firstly, we convert the sentences to vector representations and apply multilayer perceptron (MLP) to extract feature further. Then we calculate the corresponding co-attention weights for two sentences. Next, we aggregate the attention weights with the sentence representations. Finally, we use pooling to combine features and use softmax function for the final decision.

3.1 Encoding Layer

The purpose of the encoding layer is to encode the premises and the hypothesizes respectively. Encoding layer uses MLP to make the model simple and fast. After obtaining word embedding sequences $\mathbf{p}$ and $\mathbf{h}$, the features are extracted to obtain $\bar{\mathbf{p}}$ and $\bar{\mathbf{h}}$, as equations (1) to (2).

$$\bar{\boldsymbol{p}} = \delta(W\boldsymbol{p} + b) \tag{1}$$

$$\bar{\boldsymbol{h}} = \delta(W\boldsymbol{h} + b) \tag{2}$$

Where $\mathbf{p} \in \mathbf{R}^{\mathbf{m}\times\mathbf{d}}, \mathbf{h} \in \mathbf{R}^{\mathbf{n}\times\mathbf{d}}$, $\bar{\mathbf{p}} \in \mathbf{R}^{\mathbf{m}\times\mathbf{d}}$, $\bar{\mathbf{h}} \in \mathbf{R}^{\mathbf{n}\times\mathbf{d}}$, and W, b is the network parameter, $\mathbf{d}$ is the dimension of word embedding, and δ is the activation function. The lengths of $\mathbf{p}$ and $\mathbf{h}$ are $\mathbf{m}$ and $\mathbf{n}$ respectively.

3.2 Co-Attention Layer

Co-attention layer is to obtain the interactive information of two sentences through the calculation of co-attention between one sentence and another.

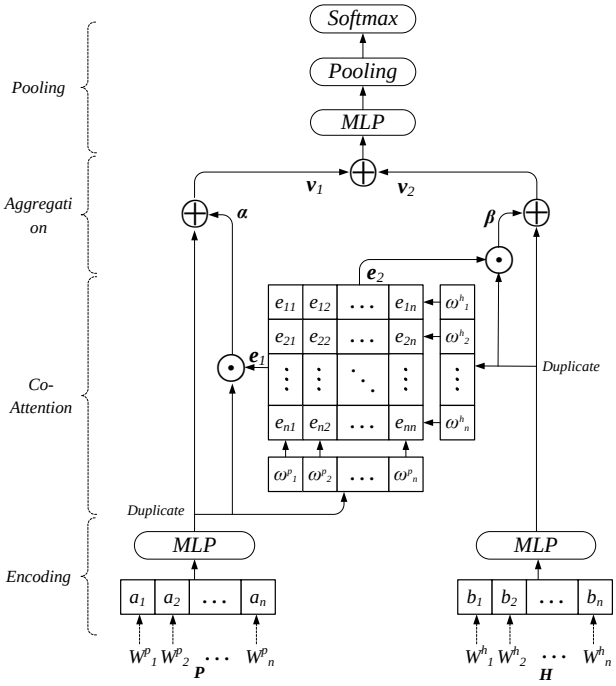


Fig. 1. Co-Attention model's structure chart. From bottom to top, **P** and **H** represent premise and hypothesis sentences of the encoding layer, respectively. MLP stands for multi-layer perceptron, Duplicate stands for copy operation, and ⊕ stands aggregation for multiple vectors, ⊙ represents the dot multiplication operation of two vectors.

Firstly, the words in sentence pairs are aligned. Then the attention weights of each sentence are calculated respectively. The alignment refers to build an m×n matrix $\boldsymbol{E}_{mn}$ with the words in two sequences $\overline{\mathbf{p}}$ and $\overline{\mathbf{h}}$ as the rows and columns respectively. Next, we use attention calculation based on $\boldsymbol{e}_{ij}$ that is the elements of $\boldsymbol{E}_{mn}$ to get the attention weights β_i ($\overline{\mathbf{h}}$ relative to $\overline{\mathbf{p}}$) and the attention weights α_j ($\overline{\mathbf{p}}$ relative to $\overline{\mathbf{h}}$). Finally, we obtain similar parts of the relationships between premises and hypothesizes. The co-attention mechanism obtains more interactive information than the self-attention, which is helpful for the judgment of the relationship between sentences. The process for calculating the weight matrix are in equations (3) to (5).

$$\mathrm{E}_{mn} = \overline{\mathbf{p}}^{T}\ \overline{\mathbf{h}} \tag{3}$$

$$\beta_i = \sum_1^n \frac{\exp(\boldsymbol{e_{ij}})}{\sum_{k=1}^{j} \exp(\boldsymbol{e_{ik}})} h_j \tag{4}$$

$$\alpha_j = \sum_1^m \frac{\exp(\boldsymbol{e_{ij}})}{\sum_{k=1}^{j} \exp(\boldsymbol{e_{jk}})} p_i \tag{5}$$

Where β_i represents attention weight that $\overline{\mathbf{h}}$ aligned with $\overline{\mathbf{p}}$; α_j attention weight that $\overline{\mathbf{p}}$ aligned with $\overline{\mathbf{h}}$.

3.3 Aggregation Layer

Aggregation layer is to aggregate the features obtained by the co-attention layer with the features of the encoding layer.

The vectors $p_i \in \overline{\mathbf{p}}$ and $h_j \in \overline{\mathbf{h}}$ are aggregated with the attention weights β_i and α_j, respectively. The method of aggregation includes concatenation, subtraction and multiplication, aim to get the results of the comparison between the sentences and their attention weights. A perceptron network layer G forwards the aggregation of each sentence. The outputs of G are the weight vectors $\boldsymbol{v_{1,i}}$ and $\boldsymbol{v_{2,j}}$ corresponding to the words of each sentence, as shown in equations (6) and (7).

$$\boldsymbol{v_{1,i}} = \mathrm{G}([\beta_i ; p_i;\ p_i - \beta_i; p_i \bullet \beta_i]) \tag{6}$$

$$\boldsymbol{v_{2,j}} = \mathrm{G}([\alpha_j ; h_j; h_j - \alpha_j; h_j \bullet \alpha_j]) \tag{7}$$

3.4 Pooling Layer

Pooling layer is to further extract the features obtained by aggregation layer. At first, the words' weight vectors $\boldsymbol{v_{1,i}}$ and $\boldsymbol{v_{2,j}}$ are accumulated into sentences' weight vectors $\boldsymbol{v_1}$ and$\boldsymbol{v_2}$, respectively. We use max pooling operation of sentences to get $\boldsymbol{v_3}$, $\boldsymbol{v_4}$ for two sentences. We connect $\boldsymbol{v_1}$, $\boldsymbol{v_2}$, $\boldsymbol{v_3}$ and $\boldsymbol{v_4}$, and feed them to a forward neural network H to obtain a vector $\mathbf{v} \in \boldsymbol{R^3}$. Finally, $\mathbf{v}$ is converted to the final label $\tilde{l}$ by softmax function. The formal representation of the pooling layer is shown as equations (8) to (13).

$$\boldsymbol{v_1} = \sum_1^m \boldsymbol{v_{1,i}} \tag{8}$$

$$\boldsymbol{v_2} = \sum_1^n \boldsymbol{v_{2,j}} \tag{9}$$

$$\boldsymbol{v_3} = Max\text{-}\mathrm{pooling}(p_1, p_2, \ldots p_m) \tag{10}$$

$$\boldsymbol{v_4} = Max\text{-}\mathrm{pooling}(h_1, h_2, \ldots h_n) \tag{11}$$

$$\mathbf{v} = \mathrm{H}([\boldsymbol{v_1}; \boldsymbol{v_2};\ \boldsymbol{v_3};\ \boldsymbol{v_4}]) \tag{12}$$

$$\tilde{l} = \mathrm{softmax}(\mathbf{v}) \tag{13}$$

4 Experiments and Analysis

4.1 Experimental Setup

In the process of data preprocessing, we use the jieba[4] word segmentation tools. Chinese word embedding are trained from the People's Daily and other corpuses by the method proposed by Li *et al.* [17]. Word embedding dimension is 300. Experiment use the PyTorch deep learning framework. The batch size is 64. The MLP hidden layer nodes are set to 300. Learning rate is 0.0004 and dropout rate is 0.3. We use the Adam function as optimization function, and use the cross entropy function as loss function. In addition, early stopping is used to prevent over-fitting. The evaluation uses accuracy which calculated as Equation (14).

$$Acc = \frac{\tilde{l}_{correct}}{l} \tag{14}$$

Where $\tilde{l}_{correct}$ represents the number of labels that are correctly classified; l is the number of true labels of raw dataset.

4.2 Experiments on Textual Entailment

We used the CCL2018 Chinese Natural Language Inference evaluation dataset. The number of training set is 90,000, development set is 10,000 and test set is 10,000. Three categories are balance in each dataset (Table 3).

Table 3. Category statistics of datasets

	Neutral	Entailment	Contradiction	Total
Train	31,325	29,738	28,937	90,000
Dev	3,098	3,485	3,417	10,000
Test	3,182	3,475	3,343	10,000

Experimental Results. We use Decomp-Att as the baseline model, and compare the CoAM with other attention models, including ESIM, SWEM, Decomp-Att, MwAtt, Self-Att, BiLSTM, and LSTM+CNN (the best accuracy model in previous evalutation) (Table 4).

The attention models such as CoAM, MwAtt and Self-Att outperformed the models without attention, which indicating that the attention model has good performance in Chinese RTE. Our model CoAM not only reaches the highest mirco-average accuracy 93.5%, but it is also effective in the classification of each category. It shows that our model is suitable for Chinese textual entailment recognition tasks than other models. Meanwhile, most attention models have the lowest accuracy rate on contradiction categories, and the highest accuracy rate is the entailment categories. It can be seen that the recognition of contradiction categories is relatively difficult.

[4] https://pypi.org/project/jieba/

Table 4. Model comparison results of CCL2018 dataset. N, E, C represent the accuracy rates in Neural, Entailment and Contradiction categories respectively. The reason that LSTM+CNN without detail categories accuracy is it was the first place in the evaluation.

Models	**Acc**	**N**	**E**	**C**
Decomp-Att (baseline)	86.5	87.1	88.0	84.4
ESIM	72.8	72.7	75.3	70.2
SWEM	74.2	74.8	75.9	72.0
LSTM+CNN	82.4	-	-	-
BiLSTM	78.3	80.0	79.3	75.6
Self-Att	90.1	90.3	91.7	88.3
MwAtt	91.3	91.8	92.5	89.6
CoAM	**93.5**	**93.8**	**94.9**	**91.7**

Results Analysis. The experimental results of CoAM show that 42.5% of all classification errors is the error of contradiction relationship recognition (Table 5 and 6). The neutral and entailment error rates are only 30.2% and 27.3%, respectively. The recognition of the entailment relationship is higher than the neutral. Because the entailment relationship has asymmetrical characteristics as a one-way inference relationship.

Table 5. Classification results confusion matrix

		Prediction		
		N	E	C
True	N	2986	107	89
	E	69	3298	108
	C	77	199	3067

Table 6. Error Rate Statistics. Number of errors is incorrect predictions in each category, proportion refers to percentage of each category of error in all errors.

Labels	Number of errors	Proportion
N	196	30.2%
E	177	27.3%
C	276	42.5%

In order to analyze the role of the attention mechanism, we outputs the attention weights between premise and hypothesis in each attention model. The relationship of the example shown in Figure 2 is contradiction. The recognition result of CoAM is correct, but Decomp-Att and ESIM are wrong.

Figure 2 shows that the attention weights are higher between similar words in models CoAM and ESIM. However, the ESIM is not effective in distinguishing the words with different meaning. The difference of Decomp-Att's attention weights for various words is lower than CoAM. For the sentences "四位芭蕾舞演员正舞台上跳舞 (Four ballet dancers are dancing on the stage)" and "所有的芭蕾舞演员都在舞台上休息 (All the ballet dancers rest on the stage)", the model CoAM and ESIM both

can recognize the same words, but the ESIM couldn't recognize different words such as "休息 (rest)" and "跳舞 (dancing)". The attention weight of "休息 (rest)" and "跳舞 (dancing)" in ESIM is higher than that in CoAM. The attention weights of the same words in Decomp-Att is lower than that in CoAM. Therefore, the CoAM is more effective in recognizing both the same words and different words than them. The experimental result show the mechanism of co-attention is more effective.

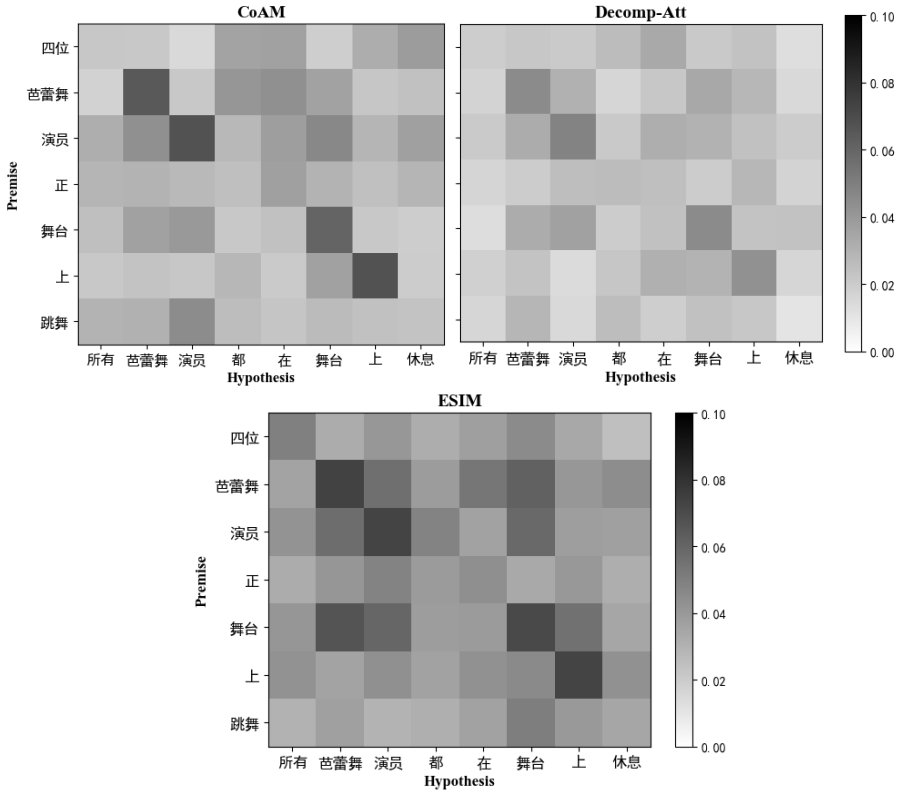
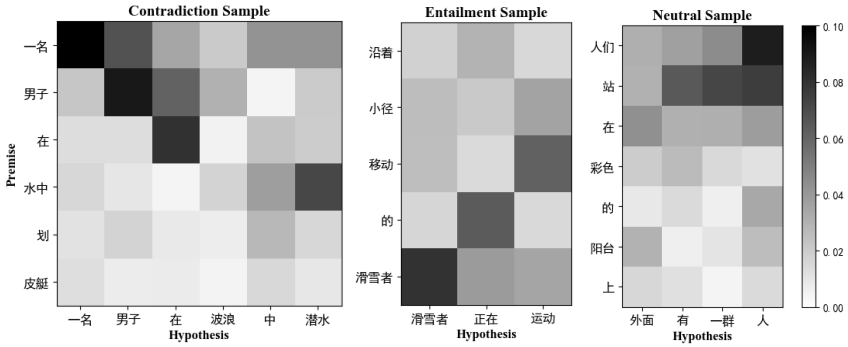


Fig. 2. Attention weight Matrix of models. The darker of the color, the greater of correlation between sentence pair.

However, the model has two problems (Figure 3): (1) Limited recognition effect on short text sentence pairs. (2) The recognition of synonyms and antonyms in the model still needs to be improved. For example, in the first example (subfigure 3.a), the model failed to obtain the semantic relationship between "皮艇 (kayak)" and "潜水 (diving)". As in the second example (subfigure 3.b), the hypothesis sentence is too short to correctly recognize their relationship. In error sample statistics, the length of 45.3% of wrong predicted examples are less than four. The third example (subfigure 3.c) shows that the deep semantic relationship is still difficult to be reflected in ordinary attention calculation.



3.a **3.b** **3.c**

Fig. 3. Attention weight Matrix of Sentence Pairs

Ablation Study. Ablation analysis shows that attention part and the subtraction operation in aggregation layer are the most important operations (Table 7).

Table 7. Model Ablation study results

Model	Acc
CoAM – co-attention	79.2
CoAM – pooling	90.2
$\text{CoAM}^{\text{Aggregation}}$ – dot	91.7
$\text{CoAM}^{\text{Aggregation}}$ – cat	89.3
$\text{CoAM}^{\text{Aggregation}}$ – min	86.3
CoAM	93.5

The co-attention part has the greatest contribution to the whole model, because the correct rate of the whole model drops by 14.28% without it, which is the biggest drop. It is reasonable that the attention operation is helpful to capture the mutual information hypothesis and the premise. After removing the pooling part, the model performance decreased by 3.4%, which was the least, indicating that the pooling mechanism contributed the least to the whole model. In the aggregation layer, the operation of removing the subtraction has the most decline, because textual entailment is to recognize one-way inference relationship, the asymmetric operation is useful. Connection in the aggregation layer is relatively more effective than dot multiplication.

4.3 Experiments on DBQA

Textual entailment recognition has a strong relevance to document-based QA (DBQA) task. The DBQA task is to give a question and an answer and judge whether the answer matches the question. It can be regarded as a binary classification problem of sentences' relationship. In order to verify the adaptability of the model for cross-task, this paper conducted experiments on the DBQA dataset of the NLPCC2016 evaluation task. The NLPCC2016 DBQA dataset has 181,882 QA pairs in training set

and 122,531 QA pairs in test set. Every QA pair is divided into two categories: correct and wrong. We use experimental implementation trained on RTE task for fine-turn. The evaluation index uses the Accuracy (ACC), the Macro Average Precision (MAP) and the Mean Reciprocal Rank (MRR).

Table 8. Model comparison result

	ACC	MAP	MRR
1(CNN)	0.7906	0.8586	0.8592
2(CNN+LSTM)	0.7385	0.8263	0.8269
3(Bi-LSTM)	0.7144	0.8111	0.8120
CoAM	0.7233	0.8174	0.7951
Dec-comp(baseline)	0.6954	0.7825	0.7632
ESIM	0.6478	0.6573	0.6733
SWEM	0.6326	0.6623	0.6815

The performance of CoAM on the NLPCC2016 DBQA test set is shown in Table 8. The model 1-3 are the top 3 models in the evaluation. The accuracy rate of our model is 72.3%, which exceeded the accuracy of the third model BiLSTM. The top two models of the evaluation combined external knowledge to improve accuracy. We also outperformed the baseline model and other attention models. Experiments demonstrated the effectiveness of the co-attention model on the QA task, and the ability of the model for cross-tasks.

5 Conclusions and Future Work

We presented a simple co-attention and aggregation based model for Chinese Recognizing Textual Entailment. The main contribution of the model is to combine the attention mechanism and the aggregation mechanism. It uses the encoding part information to improve the extraction ability of the inter-sentence information. Our model achieved the state of the art on the CCL2018 textual entailment dataset with 93.5%, and the model outperformed the other models in the recognition of contradiction categories. At the same time, the model achieved an accuracy of 72.3% on the NLPCC2016 DBQA dataset.

The next step is to improve the recognition of contradiction relationships. On the one hand, to add external knowledge to solve the problem of short sentence recognition. On the other hand, to add semantic information such as synonym or antonyms to increase the performance.

References

1. Bowman, S. R., Angeli, G., Potts, C.: A large annotated corpus for learning natural langua ge inference. Computer Science (2015).

2. Wang, S., Jiang, J.: Learning natural language inference with LSTM. arXiv preprint arXiv: 1512.08849, (2015).
3. Tan, Y., Liu, Z., Lv, X.: CNN and BiLSTM Based Chinese Textual Entailment Recognitio n. Journal of Chinese information processing, 32(7): 11-19 (2018).
4. Chen, Q., Chen, X., Guo, X.: Multiple-to-One Chinese Textual Entailment for Reading Co mprehension. Journal of Chinese information processing, 32(4): 87-94 (2018).
5. Parikh, A P., Täckström, O., Das, D.: A Decomposable Attention Model for Natural Langu age Inference.:2249-2255(2016).
6. Shen, D., Wang, G., Wang, W.: Baseline needs more love: On simple word-embedding-ba sed models and associated pooling mechanisms. arXiv preprint arXiv:1805.09843, (2018).
7. Dagan, I., Glickman, O.: Probabilistic textual entailment: Generic applied modeling of lan guage variability. Learning Methods for Text Understanding and Mining, 2004: 26-29 (20 04).
8. Dagan, I., Glickman, O., Magnini, B.: The PASCAL recognising textual entailment challe nge//Machine Learning Challenges Workshop. Springer, Berlin, Heidelberg, 177-190 (200 5).
9. Dagan, I., Roth, D., Sammons, M.: Recognizing textual entailment: Models and applicatio ns. Synthesis Lectures on Human Language Technologies, 6(4): 1-220 (2013).
10. Burger, J., Ferro, L.: Generating an entailment corpus from news headlines[C]//Proceeding s of the ACL workshop on Empirical Modeling of Semantic Equivalence and Entailment. Association for Computational Linguistics, 49-54 (2005).
11. Williams, A., Nangia, N., Bowman S R.: A broad-coverage challenge corpus for sentence understanding through inference. arXiv preprint arXiv:1704.05426 (2017).
12. Conneau, A., Kiela, D., Schwenk, H.: Supervised learning of universal sentence representa tions from natural language inference data. arXiv preprint arXiv:1705.02364 (2017).
13. Chen, Q., Zhu, X., Ling, Z.: Enhanced LSTM for Natural Language Inference. 1657-1668 (2016).
14. Tan, C., Wei, F., Wang, W.: Multiway Attention Networks for Modeling Sentence Pairs. IJCAI. 4411-4417 (2018).
15. Ghaeini R, Hasan S A, Datla V, et al. DR-BiLSTM: Dependent Reading Bidirectional LSTM for Natural Language Inference [J]. (2018).
16. Kim, S., Kang, I., Kwak, N.: Semantic Sentence Matching with Densely-connected Recurr ent and Co-attentive Information. (2018).
17. Li, S., Zhao, Z., Hu, R., Li, W., Liu, T., Du, X.: Ana-logical Reasoning on Chinese Morph ological and Semantic Re-lations. arXiv preprint arXiv:1805.06504 (2018).