ECNU at SemEval-2018 Task 12: An End-to-End Attention-based Neural Network for the Argument Reasoning Comprehension Task

Junfeng Tian[1], **Man Lan**[1,2*] and **Yuanbin Wu**[1,2]
[1] School of Computer Science and Software Engineering,
East China Normal University, Shanghai, P.R.China
[2] Shanghai Key Laboratory of Multidimensional Information Processing
51151201048@stu.ecnu.edu.cn, {mlan, ybwu}@cs.ecnu.edu.cn

Abstract

This paper presents our submissions to SemEval 2018 Task 12: the Argument Reasoning Comprehension Task. We investigate an end-to-end attention-based neural network to select the correct answer from two candidate warrants, with their surrounds (i.e., claim, reason, debate context) as attention vector. Besides, to obtain the distinguishable representations, we extract the different parts between the two warrants (claims) which have the opposite meaning, and apply them as attention vectors into another attention neural network. Our model achieves 60.4% accuracy and ranks 3rd among 22 participating systems.

1 Introduction

Reasoning is a crucial part of natural language argumentation. In order to comprehend an argument, one must analyze its *warrant*, which explains why its *claim* follows form its premises (aka *reasons*) (Habernal et al., 2018a).

SemEval-2018 Task 12 provides the argument reasoning comprehension task (Habernal et al., 2018b). Given a reason R and a claim C along with the title T and a short description I of the debate they occur in, identify the correct warrant W from two candidates: the correct warrant W and incorrect alternative warrant AW. Figure 1 gives an example. The abstract structure of an argument is *Reason* → (since) *Warrant* → (therefore) *Claim*.

The challenging factor is that both options are plausible and lexically very close while leading to contradicting claims. Here we give three examples of the two warrant options:

Ex1: A huge pandemic would (not) be a great news story.

Ex2: The role of a citizen and a supreme court justice are inseparable /separable.

Ex3: The rest of the comments can be skipped easily /make the section unreadable.

Title: Have Comment Section Failed?
Description: In recent years, many media companies have disabled them because of widespread abuse and obscenity.
Reason: Many comments are thoughtful and insightful. And since {Warrant0 | Warrant1},
Claim: Comment sections have not failed.
✓ **Warrant0**: The rest of the comments can be skipped easily.
✗ **Warrant1**: The rest of the comments make the section unreadable.

Figure 1: An example of a debate in the argument reasoning comprehension task.

We can see that the difference between the two candidates is either to add a negative word, or to use its antonym, or to pass a negative phrase. Therefore, it is important to distinguish the opposing candidate warrants.

To address the task, we proposed an end-to-end attention-based neural network. First, we use the different part as attention vector to obtain discriminative representation of the candidate warrants and the claim. Second, we use the representation of the warrants surrounds (i.e., reason, claim, debate context) as attention vector to obtain the interactive representation of the candidates.

2 Task Definition and Motivation

Formally, given an instance containing two candidate warrants (W_0, W_1) and the context around the warrants (i.e., R, C, T, I), the goal is to choose the correct warrant $y \in \{0, 1\}$, where $y = 0$ means W_0 is the correct answer, and $y = 1$ otherwise.

The network is inspired by Siamese network (Mueller and Thyagarajan, 2016). The two candidate warrants are modeled in the same structure. First, we stack a CNN and a RNN to represent each component (i.e., two options, claim, reason, debate context). That combination makes use of the best of both worlds, the spatial and temporal worlds. Second, to address different representation between the two warrants, we extract their different part as attention vector to farther separate their representation. Similarly, we apply the same strategy to obtain the representation of the claim, since the reason chains $R \rightarrow W \rightarrow C$ and $R \rightarrow AW \rightarrow \urcorner C$ both exists. Last, we put the representations of the surrounds of warrants (i.e., reason, claim, debate context) into another attention network to get the interaction representation and choose the correct warrant which satisfies the reasoning chain $R \rightarrow W \rightarrow C$.

3 System Architecture

Figure 2 illustrates our system architecture. We first extract the different part of warrant0, warrant1 and claim (see in Sec. 3.1). Then, we obtain the context representation (in Sec. 3.2) of each components in a debate instance. In particular, we adopt intra-temporal attention (in Sec. 3.3) to obtain the discriminative representations of the warrants and the claim . After that, we concatenate the representation of {reason, claim, debate context and warrant} as attention vector, and feed them into another attention-based neural network to obtain the interaction representation of the warrants. Finally, we adopt a dense layer to obtain the probability of the two candidate warrants (in Sec. 3.4).

3.1 Extract the Different Part

The two candidate warrants are lexically very close (since they often mean the opposite), thus we extract the different part between them to serve as attention vector to guide the neural network to generate discriminative representation for the warrants. To do this, we remove the longest common prefix and suffix, and let the remain part as the different part. Note that if the remain part is empty, we use the word after the prefix as the different part. For example, "A huge pandemic would $\overline{\underline{\text{(not)}}}$ be a great news story.", we would extract "be" and "not be" rather than "NONE" and "not", because only negative terms can not capture sufficient semantic information. Finally, we obtain the different (discriminative) part of W_0, W_1, denoted as Diff_W_0, Diff_W_1. Similarly, we also get the different part between the claim and its opposite, denoted as Diff_Claim.

3.2 Context Representation

To incorporate contextual information of each components in a debate, we combine Convolutional Neural Network (CNN) and Recurrent neural network (RNN) to encode the input word vectors. CNN is good at dealing with spatially related data, such as "sometimes warranted" and "rarely warranted", while RNN is good at temporal signals. Instead of using a typical vanilla RNN, we use Long Short-Term Memory Network (Hochreiter and Schmidhuber, 1997) for eliminating the issue of long term dependencies.

Given a sentence $S = \{w_i\}_1^n$, we first map each word w_i into its vector representation $x_i \in \mathbb{R}^d$ via a look-up table of word embeddings (d is the dimension of the word embeddings).

We then adopt CNN on the input sequence $\{x_i\}_1^n$ to obtain the spatial representation $\{x'_i\}_1^n$:

$$e_i^j = \text{ReLU}(w^j[x_i, \ldots, x_{i+k-1}]) \quad (1)$$

$$x'_i = [e_i^1, \ldots, e_i^m](1 \leq j \leq m) \quad (2)$$

where k is the window size, w^j is the parameter of a filter, m is the number of the filters. We also adopt padding before the convolution operation. As a result, we obtain the spatial representations $x'_i \in \mathbb{R}^m$, which has the same length as the input sequence.

After that, we utilize a bi-directional LSTM (BiLSTM) to obtain the temporal information. For each time step t, the LSTM unit computation corresponds to :

$$i_t = \sigma(W_i x'_t + U_i h_{t-1} + b_i) \quad (3)$$

$$f_t = \sigma(W_f x'_t + U_f h_{t-1} + b_f) \quad (4)$$

$$o_t = \sigma(W_o x'_t + U_o h_{t-1} + b_o) \quad (5)$$

$$\tilde{c}_t = \tanh(W_c x'_t + U_c h_{t-1} + b_c) \quad (6)$$

$$c_t = i_t \odot \tilde{c}_t + f_t \odot c_{t-1} \quad (7)$$

$$h_t = o_t \odot \tanh(c_t) \quad (8)$$

where σ is the element-wise sigmoid function, $\odot$ is the element-wise product and i_t, f_t, o_t ,c_t demote the input gate, forget gate, output gate and memory cell respectively.

3.3 Intra-Temporal Attention

Inspired from Habernal et al. (2018a), we use an intra-temporal attention function to attend over specific parts of the input sequence. This kind of attention encourages the model to generate different representation according to the attention vector. Habernal et al. (2018a) have shown that such

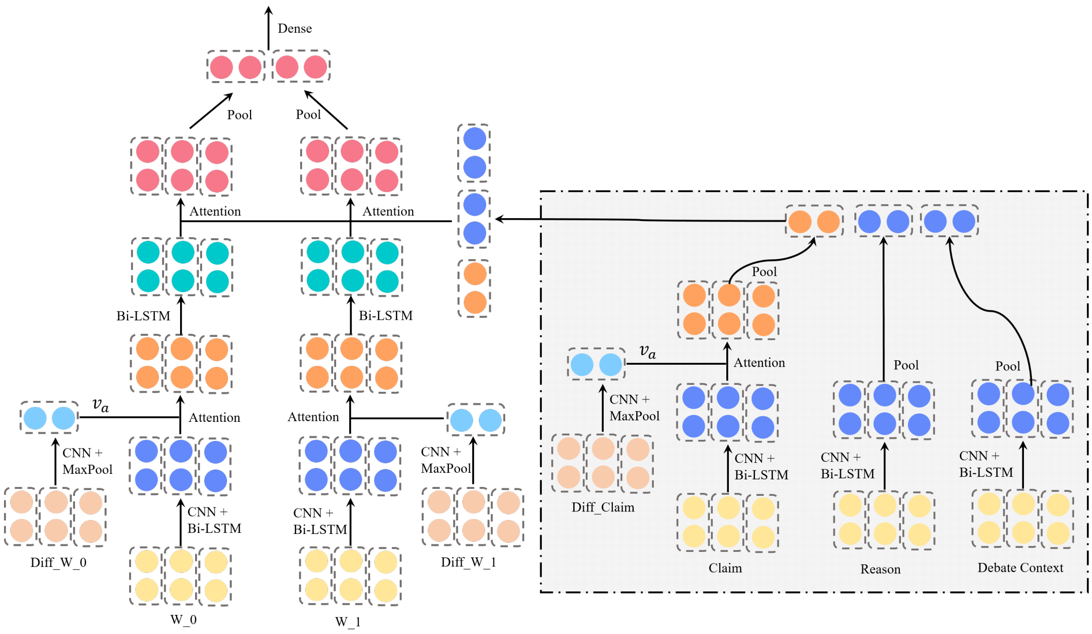

Figure 2: The system architecture

intra-temporal attention outperforms standard attention.

We define v_a as the attention vector, and h_t as the hidden states at time step t:

$$m_t = \tanh(U_m h_t \odot v_a + b_m) \tag{9}$$

$$a_t = \sigma(W_a m_t + b_a) \tag{10}$$

$$h_t = h_t \odot a_t \tag{11}$$

where a_t is the attention weights over the hidden states h_t, $\odot$ is element-wise multiplication.

We first apply the intra-temporal attention over warrant0 and warrant1, in order to obtain different warrant representations from Diff_W_0 and Diff_W_1. As a result, the model can easily distinguish the two candidate warrants. Similarly, we apply the attention over claim to make the claim representation distinguishable.

Moreover, we adopt another intra-temporal attention over warrant0 and warrant1, with the concatenation of {claim, reason, debate context} representations as attention vector. The candidate warrants receive the information from the claim, reason and debate context, and the model would select the correct warrant which satisfies the reasoning chain $R \to W \to C$.

Finally, we obtain two attended warrant vectors att_{W_0}, att_{W_1}.

3.4 Output

To evaluate the probability distribution of the two candidate warrants, we employ a feed-forward neural network with one dense layer, and apply the *softmax* function to predict the probability.

$$h_o = \mathrm{ReLU}(W_o[att_{W_0}, att_{W_1}]) \tag{12}$$

$$\hat{p} = softmax(W_p h_o) \tag{13}$$

As for the optimization, cross-entropy loss is used as the loss function since we are handling a classification problem:

$$\mathcal{L} = -\frac{1}{m}\sum_{i=1}^{m}\Big(y_i \log \hat{p}_i + (1-y_i)\log(1-\hat{p}_i)\Big) \tag{14}$$

where m is the number of training pairs.

4 Experiments

4.1 Datasets

SemEval 2018 provided 1,970 instances for the argument reasoning comprehension task (Habernal et al., 2018b). The instances are divided into three sets based on the year when the debates are taken from. Table 1 lists the statistics of the datasets. We also include the number of debate topics of each set.

Being a binary task, accuracy (Acc) is adopted as the evaluation metric.

Dataset	# Pairs	# Topics	Source Year
Train	1,210	111	2011-2015
Dev	316	31	2016
Test	444	30	2017

Table 1: The statistics of the datasets

Approach	Dev
Intra-warrant attention	0.638 (±0.024)
Intra-warrant attention w/ context	0.637 (±0.040)
Our basic model	0.666 (±0.019)
· + Diff_$\{W_0, W_1\}$	0.678 (±0.001)
· + Diff_$\{W_0, W_1\}$ + CNN	0.675 (±0.010)
· + Diff_$\{W_0, W_1$, Claim$\}$ + CNN	0.676 (±0.010)
Ensemble (Vote)	0.708

Table 2: Accuracy of each approach on the developing dataset.

4.2 Parameters Setting

The word embeddings are initialized with the 300d pre-trained word2vec (Mikolov et al., 2013), and do not fine-tune during training. The window sizes of CNN is (1,2,3) and the kernel size is 50. The dimensions of the hidden size in Bi-LSTM and Att-LSTM are set to 50. The dense layer in Output is 25. We train the model using Adam (Kingma and Ba, 2014) with gradient clipping (the max norm is set to 30, batch size is 32), The networks are regularized by dropout (the dropout ratio equals 0.8). We ran each model three times with random initializations.

4.3 Results on Training Data

Table 2 shows the results of each components of our attention-based neural network. We have the following findings:

(1) Comparing with the Intra-warrant attention (w/ context) provided by the organizer, our basic model obtains 2.8% improvement through sharing parameters in Bi-LSTM. It indicates that the neural network need sufficient training data and parameters sharing could alleviate the demand.

(2) All of the three improvements achieves improved accuracy. It suggests that utilizing the different part as attention vector can obtain discriminative representation, which is beneficial for choosing the correct answer.

(3) The introduction of CNN does not seem to improve the performance of the model. The possible reason may be that RNN actually learn any computational function and capture the spatial information.

(4) The ensemble of the three networks can further improve the performance. Therefore, we configure the ensemble model as our final submission.

Approach	Test
Human average	0.798 (±0.162)
Human w/ training in reasoning	0.909 (±0.114)
Random baseline	0.508 (±0.015)
Intra-warrant attention w/ context	0.584 (±0.015)
Rank 1: GIST	0.712
Rank 2: blcu_nlp	0.606
Rank 3: ECNU	0.604

Table 3: Accuracy of each approach (humans and systems) on the test set.

4.4 Results on Test Data

Table 3 lists the results of three top systems and several baselines provided by the organizer. We find that: (1) Our model outperforms the Intra-warrant attention w/ context model by 2% in terms of accuracy, which demonstrates the efficiency of our model. (2) Comparing with GIST and blcu_nlp, they both use the pretrained ESIM model (Chen et al., 2017) trained on SNLI (Bowman et al., 2015) and MultiNLI (Nangia et al., 2017). Our model does not require any extra resources and our result is comparable to blcu_nlp. (3) Our result is worse than GIST. The possible reason is that GIST uses ESIM to transfer common senses from NLI dataset while our model only uses the limited dataset which is insufficient to learn the model.

5 Conclusion

In this work, we propose an end-to-end neural network for the reading comprehension task. We stack a CNN and a RNN to represent each component in a debate and extract the warrants' and claim's different part as attention vector to obtain their discriminative representation. Moreover, we use another attention network to incorporate the information of reason, claim, debate context into the interactive representation of the warrants for final decisions. Our model achieves 60.4% accuracy and ranks 3rd among 22 participating systems.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui

Jiang, and Diana Inkpen. 2017. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018a. The argument reasoning comprehension task: Identification and reconstruction of implicit warrants. In *Proceedings of NAACL*.

Ivan Habernal, Henning Wachsmuth, Iryna Gurevych, and Benno Stein. 2018b. Semeval-2018 task 12: The argument reasoning comprehension task. In *Proceedings of the 12th International Workshop on Semantic Evaluation (SemEval-2018)*, New Orleans, LA, USA. Association for Computational Linguistics.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Diederik P. Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980.

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*, pages 3111–3119.

Jonas Mueller and Aditya Thyagarajan. 2016. Siamese recurrent architectures for learning sentence similarity. In *Thirtieth AAAI Conference on Artificial Intelligence*.

Nikita Nangia, Adina Williams, Angeliki Lazaridou, and Samuel Bowman. 2017. The repeval 2017 shared task: Multi-genre natural language inference with sentence representations. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, pages 1–10.