Bi-directional Gated Memory Networks for Answer Selection

Wei Wu, Houfeng Wang, Sujian Li

Key Laboratory of Computational Linguistics, Ministry of Education
School of Electronics Engineering and Computer Science, Peking University
No.5 Yiheyuan Road, Haidian District, Beijing, 100871, China
{wu.wei,wanghf,lisujian}@pku.edu.cn

Abstract. Answer selection is a crucial subtask of the open domain question answering problem. In this paper, we introduce the Bi-directional Gated Memory Network (BGMN) to model the interactions between question and answer. We match question ($\boldsymbol{P}$) and answer ($\boldsymbol{Q}$) in two directions. In each direction(for example $\boldsymbol{P} \rightarrow \boldsymbol{Q}$), sentence representation of $\boldsymbol{P}$ triggers an iterative attention process that aggregates informative evidence of $\boldsymbol{Q}$. In each iteration, sentence representation of $\boldsymbol{P}$ and aggregated evidence of $\boldsymbol{Q}$ so far are passed through a gate determining the importance of the two when attend to every step of $\boldsymbol{Q}$. Finally based on the aggregated evidence, the decision is made through a fully connected network. Experimental results on SemEval-2015 Task 3 dataset demonstrate that our proposed method substantially outperforms several strong baselines. Further experiments show that our model is general and can be applied to other sentence-pair modeling tasks.

Keywords: Question Answering, Attention Mechanism, Memory Networks

1 Introduction

Answer selection is a long-standing challenge in NLP and catches many researchers' attention. Given a question and a set of corresponding answers, the task is to classify the answers as '*Good*', '*Potential*' and '*Bad*' according to the degree to which they can answer the question. Neural network based methods have made tremendous progress in this area, one of the key factors in these achievements has been the use of attention mechanism which emphasizes specific parts of one sentence which are relevant to the other sentence.

Table 1 lists an example question and its two corresponding answers. A question usually includes a title which gives a brief summary of the question and a body which describes the question in detail. Answer 1 is a good answer, because it provides helpful information, such as *'check it to the traffic dept'*. Although Answer 2 is relevant to the question, it does not contain any useful information for the question so it is regarded as a bad answer.

From this example we can see why the attention mechanism is useful in answer selection task, one important characteristic is redundancy and noise[29] in both question and answer which may act as a distraction. In order to better model the relationship between question and answer, we must focus on the more informative parts from the question (*'check the history of the car'* in this example) and the more informative parts from the answer (*'check it to the traffic dept.'* in this example).

Question title	Checking the history of the car.
Question body	How can one check the history of the car like maintenance, accident or service history. In every advertisement of the car, people used to write "Accident Free", but in most cases, car have at least one or two accident, which is not easily detectable through Car Inspection Company. Share your opinion in this regard.
Answer 1	Depends on the owner of the car.. if she/he reported the accident/s i believe u can check it to the traffic dept.. but some owners are not doing that especially if its only a small accident.. try ur luck and go to the traffic dept..
Answer 2	How about those who claim a low mileage by tampering with the car fuse box? In my sense if you're not able to detect traces of an accident then it is probably not worth mentioning...For best results buy a new car :)

Table 1. an example question and answers for answer selection from SemEval-2015 Task 3 English dataset

Attention mechanisms in most prior works typically have one of these limitations: First, they only match question to answer but neglecting the other direction. thus, they can not neglect useless segments from a potentially long question like the above example. Second, they only use a single-iteration attention mechanism which may not find the information useful enough to determine the answer quality. In the above example, single-iteration attention may find *car*, *detect*, *accident* to be relevant in answer 2 with the question thus making a wrong decision.

In this paper, to tackle these limitations, we introduce the Bi-directional Gated Memory Network (BGMN), an end-to-end neural network for answer selection. We use bi-directional attention mechanism to extract useful information from both directions. In order to refine attention representation iteratively we adopt the mechanism of revisiting question and answer multiply times. Furthermore, to improve the performance of memory mechanism in this task, an additional gate is added to determine the relative importance between the memory of one sentence and the representation of the other sentence, thus obtain a more focused relevance vector which can be used both in attention and formation of memory vector. Like the gating mechanism in LSTM[10] that optionally let information through cell state, this additional gate can control the extent to which the memory of one sentence and the representation of the other sentence can flow into the next iteration and generate the memory representation.

Our model consists of three parts: 1) the recurrent network to encode question and answer separately, 2) the gated memory network to iteratively aggregate evidence that is useful for answer selection, 3) the fully connected network to estimate the probability of labels representing the relationship between question and answer. The main contribution of our work can be summarized as follows:

- We apply the memory mechanism which can iteratively aggregate evidence from both directions to the answer selection task.
- We add an additional gate to memory networks to account for the fact that the memory of one sentence and the representation of the other sentence are of different importance when used in attention.
- Our proposed model yields state-of-the-art result on data from SemEval-2015 Task3 on Answer Selection in Community Question Answering.
- Our model achieves competitive result on the Stanford Natural Language Inference (SNLI) corpus demonstrating its effectiveness in the overall sentence-pair modeling task.

2 Related Works

2.1 Answer Selection

Answer selection task has been widely studied by many previous work. The methods using statistic classifiers ([28], [24], [18]) rely heavily on feature engineering, linguistic tools or external resources. While these methods show effectiveness, they might suffer from the availability of additional resources and errors of many NLP tools. Recently there are many works using deep learning architecture to represent the question and answer in the same hidden space, and then the task can be converted into a classification or learning-to-rank problem using these hidden representations. Among them, [8] models question and answer separately with multi-layer CNN, [22] proposes an attention-based RNN model which introduces question attention to answer representation. Simple as their model may be, they have not consider the interaction between question and answer thus only match question to answer but neglect the other direction. The single iteration attention mechanism also may not find relevant information to determine the relationship between question and answer.

2.2 Attention and Memory

A recent trend in deep learning research is the application of attention and memory mechanism. Attentive neural networks have been proved to be useful in a wide range of tasks ranging from machine translation [2], reading comprehension [9,27,17], and sentence summarization [16]. The idea is that instead of encoding each sentence as a fixed-length vector, we can focus on useful segments of text and neglect meaningless segments [22].

Memory network is a new class of attention model which can reason with inference components combined with a long-term memory component. It is first proposed in [25] where they use a memory component to answer questions via chaining facts. Despite being an effective system, their model requires that supporting facts to be labeled during training. In view of this defect, [21] proposes a memory network model that is end-to-end trainable. Their model is similar to the attention mechanism only that it makes multiple hops over the memory. [12] and [27] propose the dynamic memory network which is a general architecture for a variety of applications, including text classification, question answering, sequence modeling and visual question answering. In addition to the single-direction attention discussed above, one important defect of all these models is that they treat the memory of one sentence and the representation of the other sentence equally when used in attention and formation of memory vector. we argue that adding a gate for these two vectors can force the network to focus on the more important one, thus improving the effectiveness of the memory mechanism.

3 Method

In this section, we describe the architecture of our Bi-directional Gated Memory Network (BGMN) in detail. For notation, we denote scalars with italic lower-case (e.g. s_t^i), vectors with bold lower-case (e.g. $\boldsymbol{w}_t^p$), matrices with bold upper-case (e.g. $\boldsymbol{H}_p$) and sets with cursive upper-case (e.g. $\mathcal{Y}$). We assume words have already been converted to one-hot vectors. For answer selection, we are given a question $\boldsymbol{P}$ and an answer $\boldsymbol{Q}$, where $\boldsymbol{P} = \{\boldsymbol{w}_t^p\}_{t=1}^m$ is a sentence with length m, $\boldsymbol{Q} = \{\boldsymbol{w}_t^q\}_{t=1}^n$ is a sentence with length n, our task is to predict a label $y \in \mathcal{Y}$ representing the relationship between $\boldsymbol{P}$ and $\boldsymbol{Q}$, $\mathcal{Y} = \{good, potential, bad\}$ where *good* indicates $\boldsymbol{Q}$ is definitely relevant to $\boldsymbol{P}$, *potential* indicates $\boldsymbol{Q}$ is potentially useful to $\boldsymbol{P}$, *bad* indicates $\boldsymbol{Q}$ is bad or irrelevant to $\boldsymbol{P}$. Our model estimates the conditional probability distribution $Pr(y|\boldsymbol{P}, \boldsymbol{Q})$ through the following modules.

3.1 Sentence Encoder

Consider two sentences $\boldsymbol{P} = \{\boldsymbol{w}_t^p\}_{t=1}^m$ and $\boldsymbol{Q} = \{\boldsymbol{w}_t^q\}_{t=1}^n$. We first convert words to their respective word embeddings ($\{\boldsymbol{d}_t^p\}_{t=1}^m$ and $\{\boldsymbol{d}_t^q\}_{t=1}^n$), and then use a bi-directional RNN to incorporate contextual information into the representation of each time step of $\boldsymbol{P}$ and $\boldsymbol{Q}$ respectively, The output at each time step is the concatenation of the two output vectors from both directions, i.e. $\boldsymbol{h}_t = \overrightarrow{\boldsymbol{h}_t} \| \overleftarrow{\boldsymbol{h}_t}$. the representation of each sentence ($\boldsymbol{v}_p$ and $\boldsymbol{v}_q$) is formed by the concatenation of the last vectors on both directions ($\boldsymbol{v}_p = \overrightarrow{\boldsymbol{h}_m^p} \| \overleftarrow{\boldsymbol{h}_1^p}$, $\boldsymbol{v}_q = \overrightarrow{\boldsymbol{h}_n^q} \| \overleftarrow{\boldsymbol{h}_1^q}$):

$$\boldsymbol{h}_t^p = BiRNN(\boldsymbol{h}_{t-1}^p, \boldsymbol{d}_t^p) \tag{1}$$

$$\boldsymbol{h}_t^q = BiRNN(\boldsymbol{h}_{t-1}^q, \boldsymbol{d}_t^q) \tag{2}$$

3.2 Bi-directional Gated Memory Network

This is the core layer within our model. The goal of this module is to iteratively refine the memory of each sentence with newly relevant information about that sentence. The memory of one sentence means the informative evidence about that sentence when used for determining sentence-pair relationship, it can be iteratively refined using attention mechanism. It was initialized with the representation of that sentence ($\boldsymbol{m}_p^0 = \boldsymbol{v}_p, \boldsymbol{m}_q^0 = \boldsymbol{v}_q$). In each iteration, as is shown in Figure 1, we attend the two sentences $\boldsymbol{P}$ and $\boldsymbol{Q}$ in two directions: from $\boldsymbol{P}$ to $\boldsymbol{Q}$ and from $\boldsymbol{Q}$ to $\boldsymbol{P}$. In each direction, for example $\boldsymbol{P} \rightarrow \boldsymbol{Q}$, we add an additional gate to determine the importance of the memory of one sentence and the representation of the other sentence when used to attend, thus obtaining the relevance vector $\boldsymbol{r}_q^i$ for sentence $\boldsymbol{Q}$ in iteration i:

$$\boldsymbol{r}_q^i = sigmoid\left(\boldsymbol{W}_r\left[\boldsymbol{v}_p, \boldsymbol{m}_q^i\right]\right) \odot \left[\boldsymbol{v}_p, \boldsymbol{m}_q^i\right] \tag{3}$$

We then use an attention mechanism similar to [9], the attentional representation $\boldsymbol{e}_q^i$ of sentence $\boldsymbol{Q}$ in iteration i is formed by a weighted sum of outputs from the above sentence encoder layer $\boldsymbol{H}^q = \{\boldsymbol{h}_t^q\}_{t=1}^n$, these normalized weights $\boldsymbol{s}^i$ are interpreted as the degree to which the network attends to a particular token in the answer when answering the question in iteration i:

$$\begin{aligned} \boldsymbol{n}_t^i &= tanh\left(\boldsymbol{W}_n\left[\boldsymbol{h}_t^q, \boldsymbol{r}_q^i\right]\right) \\ s_t^i &= softmax\left(\boldsymbol{W}_s \boldsymbol{n}_t^i\right) \\ \boldsymbol{e}_q^i &= \boldsymbol{H}^q \odot \boldsymbol{s}^i \end{aligned} \tag{4}$$

Finally, following [27], we use a ReLU layer to update memory with newly relevant information from iteration i:

$$\boldsymbol{m}_q^{i+1} = ReLU\left(\boldsymbol{W}_m\left[\boldsymbol{e}_q^i, \boldsymbol{r}_q^i\right]\right) \tag{5}$$

Above describes one iteration of the BGMN model, it can be applied multiple times to aggregate more information required to determine the relationship between the sentence-pair. The number of iterations is a hyper-parameter to be tuned on the development set. Empirically three or four iterations can result in good performance.

3.3 Output Layer

This layer is employed to evaluate the conditional probability distribution $Pr(y|\boldsymbol{P}, \boldsymbol{Q})$ given memory $\boldsymbol{m}_p$ and $\boldsymbol{m}_q$ from the last iteration. For that purpose, we use a two layer fully-connected neural network and apply the *softmax* function in the last layer.

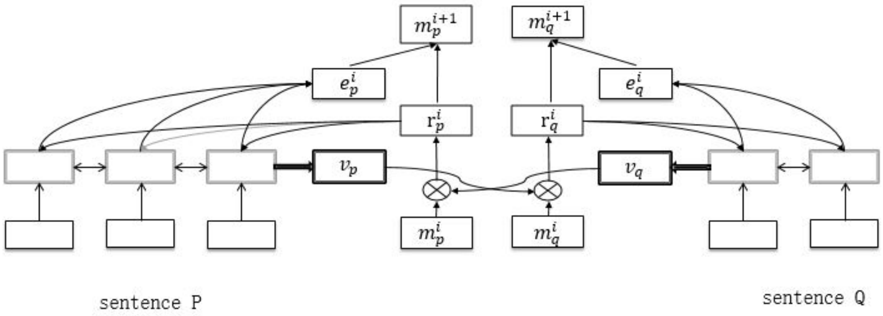

Fig. 1. Illustration for one iteration of our Bi-directional Gated Memory Network

4 Experiments

In this section, we evaluate our BGMN model on the SemEval-2015 cQA dataset. We will first introduce the basic information about this dataset in subsection 4.1 and the general setting of our model in subsection 4.2. Then we compare our model with state-of-the-art models in subsection 4.3 and demonstrate the properties of our model through some ablation study in subsection 4.4. Finally, since our BGMN model essentially models the relationship between sentences, we also test its effectiveness on another sentence-pair modeling task: textual entailment recognition in subsection 4.5.

4.1 Dataset Description

We conduct experiments on subtask A of SemEval-2015 task 3 [1]: Answer Selection in Community Question Answering to validate the effectiveness of our model. The corpus contains data from the *QatarLiving* forum[1], and is publicly available on the task's website[2]. The dataset consists of questions and a list of answers for each question. Every question consist of a short title and a more detailed description. There are also some metadata associated with them, e.g., user ID, date of posting, the question category. We do not use these metadata because we think raw texts from question and answer are enough to determine the relationship between these two sentences. Answers are required to be classified as *Good*, *Bad*, or *Potentially relevant* with respect to the question. Some statistics about the dataset are shown in Table 2.

The performance is measured by two metrics in official scorer[3]: Macro-averaged F1 and accuracy.

[1] http://www.qatarliving.com/forum
[2] http://alt.qcri.org/semeval2015/task3/
[3] http://alt.qcri.org/semeval2015/task3/data/uploads/semeval2015-task3-english-arabic-scorer.zip

Category	**Train**	**Dev**	**Test**
Questions	2,600	300	329
Answers	16,541	1,645	1,976
- *Good*	8,069	875	997
- *Potential*	1,659	187	167
- *Bad*	6,813	583	812

Table 2. Statistics of the SemEval-2015 cQA English dataset.

4.2 Experiment Setup

We use the tokenizer from NLTK [4] to preprocess each sentence. All word embeddings in the sentence encoder layer are initialized with the 300-dimensional GLoVe [14] word vectors trained on Wikipedia 2014 + Gigaword 5 and embeddings for out-of-vocabulary words are set to zero. Gated Recurrent Unit (GRU)[6] is chosen in our experiment because it performs similarly to LSTM [10] but is computationally cheaper. The hidden size of GRU is set to 200. To prevent our model from overfitting, a dropout [20] rate of 0.2 is used for all GRU layers and the fully-connected network before softmax. We use ADAM [11] for optimization with a first momentum coefficient of 0.9 and a second momentum coefficient of 0.999. We perform a small grid search over combinations of initial learning rate [1E-6, 3E-6, 1E-5], L2 regularization parameter [1E-7, 3E-7, 1E-6] and number of iterations [2, 3, 4]. We take the best configuration based on performance on the validation set, and only evaluate that configuration on the test set. In order to mitigate class imbalance problem, we use median frequency balancing as in [7] to reweight each class in the cross-entropy loss, thus the rarer a class is in training set, the larger weight it will get in the cross entropy loss.

4.3 Model Comparison

In order to analyze the performance of our BGMN model more precisely, we compare with some baselines and state-of-the-art models on this dataset. These models are introduced as follows:

- **Baseline: always 'Good'** is a basic baseline method, which assigns the most common label in training set(in this case *Good*) to every answer in the test set.
- **Baseline: BiLSTM** encodes question and answer separately with Bi-directional LSTM and sentence vectors are generated by the last hidden states from both directions,two sentence vectors are passed through a fully-connected layer to determine the sentence-pair relationship.
- **Baseline: BiLSTM-attention** resembles **Baseline: BiLSTM** but sentence vectors are generated by the attentive pooling of all hidden states[22].

- **JAIST**[23] ranks first in the evaluation of this SemEval task, it used a supervised feature rich approach, which includes topic models and word vector representation, with an SVM classifier.
- **R-CNN**[31] applies CNN to learn the joint representation of question-answer pair firstly, and then uses the joint representation as input of LSTM to learn the answer sequence of a question for labeling the matching quality of each answer.
- **KEHNN**[26] uses question categories as prior knowledge to help identify useful information and filter out noise in order to match long text.

Table 3 shows the performances of above models and our model. The results of **JAIST**, **R-CNN** and **KEHNN** are from original papers. **Baseline: always 'Good'** is the worst because it did not use any information from the test set. **Baseline: BiLSTM** performs quite well in terms of accuracy demonstrating its strong power in modeling sequences. **Baseline: BiLSTM-attention** performs better in terms of Macro-F1 but worse in terms of accuracy than **Baseline: BiLSTM**. A closer look into the result show that **Baseline: BiLSTM-attention** can better model the samples in relatively less class '*Potential*' which improves the Macro-average result but may hurt overall accuracy. The more complicated **R-CNN** is only slightly better than **Baseline: BiLSTM-attention** and **KEHNN** is only slightly better than **BiLSTM**, demonstrating that there are much room to be improved. We can see that our model achieves the state-of-the-art performance for this community question answering task despite its simplicity over **R-CNN** and **KEHNN**.

Models	**Macro F1**	**Accuracy**
baseline:always 'Good'	22.36	50.46
baseline:BiLSTM	51.94	74.75
baseline:BiLSTM-attention	55.61	71.26
JAIST[23]	57.19	72.67
R-CNN[31]	56.14	-
KEHNN[26]	-	74.8
Our BGMN model	**58.55**	**75.81**

Table 3. Results on the SemEval-2015 cQA English dataset.

4.4 Ablation Study

In this subsection, we conduct a series of studies to evaluate the effectiveness of each model features. We build several ablation models by removing one feature at a time. Table 4 shows the performance of all ablation models and our full BGMN model on the SemEval-2015 Task 3 dataset. We can see that removing any component from the BGMN model decreases the performance significantly. Removing question-to-answer attention induces more performance loss than removing answer-to-question attention, demonstrating question-to-answer attention is more important in answer selection task. Among all the features, gating is the most crucial feature for our full model to achieve good performance. When we set the number of iterations to 1 in our model, accuracy drops significantly, demonstrating the necessity of the memory component.

Models	**Accuracy**
w/o question-to-answer attention	74.13
w/o answer-to-question attention	74.80
w/o gating[4]	71.82
w/o memory	73.15
Our BGMN model	**75.81**

Table 4. Ablation study on the SNLI test set.

4.5 Further Study on Sentence-Pair Modeling

Our model can achieve state-of-the-art result on answer selection task, but due to the nature of our model is classification of relationship between a pair of sentences, we also experiment on textual entailment recognition task. Experiment results show the effectiveness of our model for this task.

Recognizing Textual Entailment Recognizing Textual Entailment is essential in tasks ranging from information retrieval to semantic parsing to commonsense reasoning. For natural language inference task, $\boldsymbol{P}$ is a premise sentence, $\boldsymbol{Q}$ is a hypothesis sentence, and $\mathcal{Y} = \{entailment, neural, contradiction\}$ where *entailment* indicates $\boldsymbol{Q}$ can be inferred from $\boldsymbol{P}$, *neural* indicates $\boldsymbol{P}$ and $\boldsymbol{Q}$ are irrelevant, *contradiction* indicates $\boldsymbol{Q}$ can not be true condition on $\boldsymbol{P}$. Previous works often stuck in employing engineered NLP pipelines, extensive manual creation of features, as well as various external resources (e.g. [13], [30], [3]). [15] proposes a sentence-by-word attentive LSTM model that reads two sentences to determine entailment, and extended that model with a word-by-word neural attention mechanism that encourages reasoning over entailments of pairs of words and phrases. However, its sentence-by-word model performs poorly and word-by-word model is computational expensive. Despite being a much simpler attention mechanism, our proposed model still outperforms their word-by-word attentive model by more than 1 point.

[4] We just set relevance to be the concatenation of two input vectors.

Experiments on Textual Entailment Recognition In this subsection, we conduct experiments on the SNLI dataset[5]. It is a large corpus with over 55K training sentence pairs and its labels are more balanced and is publicly available[5]. Table 5 shows the performances of some competitive models and our model. All baseline results are from original paper. We can see that the performance of our model is on par with some state-of-the-art models. Especially when compared with [15], our model is much more concise than their word-by-word attention model but achieves a more impressive result. Therefore, our model is also effective for natural language inference task.

Models	**Accuracy**
100D LSTM encoders[5]	77.6
Attention, two-way[15]	82.4
Word-by-word attention[15]	83.5
MKAL[19]	84.2
Our BGMN model	84.15

Table 5. Results on the SNLI test set.

5 Conclusion

We propose the Bi-directional Gated Memory Network(BGMN), an end-to-end neural network architecture for answer selection. Our model uses an iterative process to aggregate more relevant information which is useful to identify the relationship between question and answer. Experiment results show that our model achieves state-of-the-art performance in SemEval-2015 cQA task. Ablation study show that all features of our model are crucial for good performance. Further experiments on the SNLI dataset demonstrate that our model is general and can be applied to more sentence-pair modeling tasks. Future work involves incorporating label dependency in answers of the same question in answer selection and extend our model to a more suitable ranking-based answer selection task.

Acknowledgement

Our work is supported by National Natural Science Foundation of China (No.61370117, No.61433015 & No.61572049).

[5] `https://nlp.stanford.edu/projects/snli/`

References

1. PreslavNakov LluısMarquez WalidMagdy AlessandroMoschitti, James Glass, and Bilal Randeree. Semeval-2015 task 3: Answer selection in community question answering. *SemEval-2015*, 269, 2015.
2. Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *CoRR*, abs/1409.0473, 2014.
3. Islam Beltagy, Stephen Roller, Pengxiang Cheng, Katrin Erk, and Raymond J Mooney. Representing meaning with a combination of logical form and vectors. *arXiv preprint arXiv:1505.06816*, 2015.
4. Steven Bird. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 69–72. Association for Computational Linguistics, 2006.
5. Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *EMNLP*, 2015.
6. Kyunghyun Cho, Bart van Merrienboer, and year=2014 aglar Gülehre and Dzmitry Bahdanau and Fethi Bougares and Holger Schwenk and Yoshua Bengio, booktitle=EMNLP. Learning phrase representations using rnn encoder-decoder for statistical machine translation.
7. David Eigen and Rob Fergus. Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2650–2658, 2015.
8. Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. Applying deep learning to answer selection: A study and an open task. In *ASRU*, 2015.
9. Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701, 2015.
10. Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.
11. Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
12. Ankit Kumar, Ozan Irsoy, Peter Ondruska, Mohit Iyyer, James Bradbury, Ishaan Gulrajani, Victor Zhong, Romain Paulus, and Richard Socher. Ask me anything: Dynamic memory networks for natural language processing. In *ICML*, 2016.
13. Alice Lai and Julia Hockenmaier. Illinois-lh: A denotational and distributional approach to semantics. *Proc. SemEval*, 2:5, 2014.
14. Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *EMNLP*, volume 14, pages 1532–1543, 2014.
15. Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomas Kocisky, and Phil Blunsom. Reasoning about entailment with neural attention. In *International Conference on Learning Representations (ICLR)*, 2016.
16. Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *EMNLP*, 2015.
17. Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

18. Aliaksei Severyn and Alessandro Moschitti. Automatic feature engineering for answer selection and extraction. In *EMNLP*, volume 13, pages 458–467, 2013.
19. Lei Sha, Sujian Li, Baobao Chang, and Zhifang Sui. Recognizing textual entailment via multi-task knowledge assisted lstm. In *China National Conference on Chinese Computational Linguistics*, pages 285–298. Springer, 2016.
20. Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.
21. Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015.
22. Ming Tan, Bing Xiang, and Bowen Zhou. Lstm-based deep learning models for non-factoid answer selection. *CoRR*, abs/1511.04108, 2015.
23. Quan Hung Tran, Vu Tran, Tu Vu, Minh Nguyen, and Son Bao Pham. Jaist: Combining multiple features for answer selection in community question answering. In *Proceedings of the 9th International Workshop on Semantic Evaluation, SemEval*, volume 15, pages 215–219, 2015.
24. Mengqiu Wang, Noah A Smith, and Teruko Mitamura. What is the jeopardy model? a quasi-synchronous grammar for qa. In *EMNLP-CoNLL*, volume 7, pages 22–32, 2007.
25. Jason Weston, Sumit Chopra, and Antoine Bordes. Memory networks. *arXiv preprint arXiv:1410.3916*, 2014.
26. Yu Wu, Wei Wu, Zhoujun Li, and Ming Zhou. Knowledge enhanced hybrid neural network for text matching. *arXiv preprint arXiv:1611.04684*, 2016.
27. Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.
28. Wen-tau Yih, Ming-Wei Chang, Christopher Meek, Andrzej Pastusiak, Scott Wen-tau Yih, and Chris Meek. Question answering using enhanced lexical semantic models. 2013.
29. Xiaodong Zhang, Sujian Li, Lei Sha, and Houfeng Wang. Attentive interactive neural networks for answer selection in community question answering. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.
30. Jiang Zhao, Tian Tian Zhu, and Man Lan. Ecnu: One stone two birds: Ensemble of heterogenous measures for semantic relatedness and textual entailment. *Proceedings of the SemEval*, pages 271–277, 2014.
31. Xiaoqiang Zhou, Baotian Hu, Qingcai Chen, Buzhou Tang, and Xiaolong Wang. Answer sequence learning with neural networks for answer selection in community question answering. In *ACL*, 2015.