Attention-over-Attention Neural Networks for Reading Comprehension

Yiming Cui[†], Zhipeng Chen[†], Si Wei[†], Shijin Wang[†], Ting Liu[‡] and **Guoping Hu[†]**

[†]iFLYTEK Research, China

[‡]Research Center for Social Computing and Information Retrieval,
Harbin Institute of Technology, China

[†]{ymcui,zpchen,siwei,sjwang3,gphu}@iflytek.com
[‡]tliu@ir.hit.edu.cn

Abstract

Cloze-style queries are representative problems in reading comprehension. Over the past few months, we have seen much progress that utilizing neural network approach to solve Cloze-style questions. In this paper, we present a novel model called attention-over-attention reader for the Cloze-style reading comprehension task. Our model aims to place another attention mechanism over the document-level attention, and induces "attended attention" for final predictions. Unlike the previous works, our neural network model requires less pre-defined hyper-parameters and uses an elegant architecture for modeling. Experimental results show that the proposed attention-over-attention model significantly outperforms various state-of-the-art systems by a large margin in public datasets, such as CNN and Children's Book Test datasets.

1 Introduction

To read and comprehend the human languages are challenging tasks for the machines, which requires that the understanding of natural languages and the ability to do reasoning over various clues. Reading comprehension is a general problem in the real world, which aims to read and comprehend a given article or context, and answer the questions based on it. Among various types of reading comprehension problems, the Cloze-style queries are the fundamental ones and has become a starter in tackling machine comprehensions. The Cloze-style queries (Taylor, 1953) is the fundamental one in reading comprehensions, which shares most of the characteristics of reading comprehension, but the answer is a single word in the document.

Among various techniques, the attention-based neural network (Bahdanau et al., 2014) has become a stereotype in most of the NLP research, which is well-known by its capability of learning the "importance" distribution over the inputs by itself. To teach the machine to do Cloze-style reading comprehensions, large-scale training data is necessary for learning relationships between the given document and query. In order to create large-scale training data for neural networks, Hermann et al. (2015) released the CNN/Daily Mail news corpus, where the document is formed by the news articles and the query is extracted from the summary. Hill et al. (2015) released the Children's Book Test (CBT) dataset afterwards, where the training samples are generated from consecutive 20 sentences from books, and the query is formed by 21st sentence. Also, Cui et al. (2016) has released the Chinese reading comprehension datasets, including a human-made out-of-domain test set for future research. All previous works are focusing on automatically generating large-scale training data for neural network training, which demonstrate its importance. Furthermore, the more complicated problems the more data is needed to learn comprehensive knowledge from it, such as reasoning over multiple sentences etc.

In this paper, we present a novel and elegant neural network architecture, called *attention-over-attention* model. As we can understand the meaning literally, our model aims to place another attention mechanism on the existing document-level attention. Unlike the previous works, that are using heuristic merging functions (Cui et al., 2016), or setting various hyper-parameters (Trischler et al., 2016), our model could automatically generate an "attended attention" over the various document-level attentions,

---

and make a mutual look not only from *query-to-document* but also *document-to-query*, which will benefit from the interactive information.

To sum up, the main contributions of our work are listed as follows.

- To our knowledge, this is the first time that the *attention-over-attention* mechanism is introduced.

- Unlike the previous works on introducing complex architectures or many non-trainable hyper-parameters to the model, our model is simple and without any burdens on hyper-parameter tuning, but outperforms various state-of-the-art systems by a large margin on the public datasets.

- As our model is generalized, we believe the *attention-over-attention* mechanism can be used to other tasks as well.

The following of the paper will be arranged as follows. In Section 2, we will give an introduction to the Cloze-style reading comprehension task, and then talk about the related public datasets. Then the proposed attention-over-attention Reader will be presented in detail in Section 3. The experimental results will be given in Section 4. Related work will be discussed in Section 5. Finally, we will give a conclusion of this paper, and envision on future work.

2 Cloze-style Reading Comprehension

In this section, we will give a brief introduction to the Cloze-style reading comprehension task at the beginning. And then, several existing public datasets will be described in detail.

2.1 Task Description

As we have briefly introduced in the Section 1, the Cloze-style queries are the simple ones in the reading comprehension, and much progress has been made in recent research. Formally, a general Cloze-style query can be illustrated as a triple:

$$\langle \mathcal{D}, \mathcal{Q}, \mathcal{A} \rangle$$

The triple consists of a documen $\mathcal{D}$, a query $\mathcal{Q}$ and the answer to the query $\mathcal{A}$. Note that, the answer is usually a *single* word in the document, which requires the human to exploit contextual informations by both document and query. The type of the answer word varies from predicting a preposition given a fixed collocation to identifying a named entity from a factual illustration.

2.2 Existing Public Datasets

Large-scale training data is essential for training neural networks. Recently, several public datasets for the Cloze-style reading comprehension has been published.

CNN/Daily Mail.[1] Hermann et al. (2015) have firstly published two datasets: CNN and Daily Mail news data. They construct these datasets with web-crawled CNN and Daily Mail news data. One of the characteristics of these datasets is that the news article is often associated with a summary. So they first regard the main body of the news article as the *Document*, and the *Query* is formed by the summary of the article, where one entity word is replaced by a special placeholder to indicate the missing word. The replaced entity word will be the *Answer* of the *Query*. Apart from the release of dataset, they also proposed a methodology that anonymize the named entity tokens in the data, and the these tokens are also re-shuffle in each sample. The motivation is that the news articles are containing limited named entities, which are usually celebrities, and the world knowledge can be learned from the dataset. So this methodology aims to exploit general relationships between anonymized named entities with a single document rather than common knowledge. The following research on these datasets showed that the entity word anonymization is not that effective than expected. To have a better understanding, we show an example of CNN news dataset in the below.

---

[1]The pre-processed CNN and Daily Mail datasets are available at `http://cs.nyu.edu/~kcho/DMQA/`

Document

@entity2 celebrates his late strike as @entity5 retained their unbeaten @entity4 record with a 1 - 1 draw against @entity9 . @entity9 captain @entity14 had given the visitors a deserved first - half leadbut @entity2 's strike two minutes from time maintained @entity5 's nine - point lead at the top . @entity9 needed to win to breathe new life into the title race but they were dealt a cruel blow as @entity25 defender @entity24 was sent - off late on for two bookings in quick succession.

......

@entity92 are up to fourth after they defeated @entity93 1 - 0 at home thanks to a goal from @entity96 forward @entity97 . e-mail to a friend

Query

the result keeps *@placeholder* nine points clear and retains their unbeaten league run

Answer

@entity5

Children's Book Test. [2] There was also a dataset called the Children's Book Test (CBTest) released by Hill et al. (2015), which is built on the children's book story through Project Gutenberg. Different from the CNN/Daily Mail datasets, there is no summary available in the children's book. So they proposed another way to extract query from the original data. The document is composed by 20 consecutive sentences in the story, and the 21st sentence is regarded as the query, where one word is blanked with a special placeholder. In the CBTest datasets, there are four types of sub-datasets available which is classified by the part-of-speech tag of the answer word, containing Named Entities (NE), Common Nouns (CN), Verbs and Prepositions. In their studies, they have found that the answering of verbs and prepositions are relatively less dependent with the content of document and query, and the humans can even do preposition blank-filling without presence of document. As the aim of reading comprehension is to exploit general structure knowledge, the most of the following studies are only focusing on the NE and CN datasets.

3 Attention-over-Attention Reader

In this section, we will give a detailed introduction to the proposed Attention-over-Attention Reader (AoA Reader). Our model is primarily motivated by Kadlec et al., (2016), which aims to directly estimate the answer from the document-level attention instead of calculating blended representations of the document. As previous studies by Cui et al. (2016) showed that the investigation of query representation is necessary, and it should be paid more attention on utilizing the information of query. In this paper, we propose a novel work that placing another attention over the primary attention, to indicate the "importance" of each attentions.

Now, we will give a formal description of our proposed model. When a Cloze-style training triple $\langle \mathcal{D}, \mathcal{Q}, \mathcal{A} \rangle$ is given, the proposed model will be constructed in the following steps.

Contextual Embedding. We first transform every word in the document $\mathcal{D}$ and query $\mathcal{Q}$ into one-hot representations, and then convert them into continuous representations with a shared embedding matrix W_e. The motivation of using shared embedding weights is that, the length of query is shorter than the document, and thus the embedding weights will not be fully learned by only using a small amount of training data. By embedding sharing, both the document and query can participate in the learning of embedding, and will benefit from each other. After that, we use two bi-directional RNNs to get contextual representations of the document and query individually, where representation of each word is formed by concatenating the forward and backward hidden states. After making a trade-off between model performance and training complexity, we choose the bi-directional Gated Recurrent Unit (GRU) (Cho et al., 2014) in our implementation.

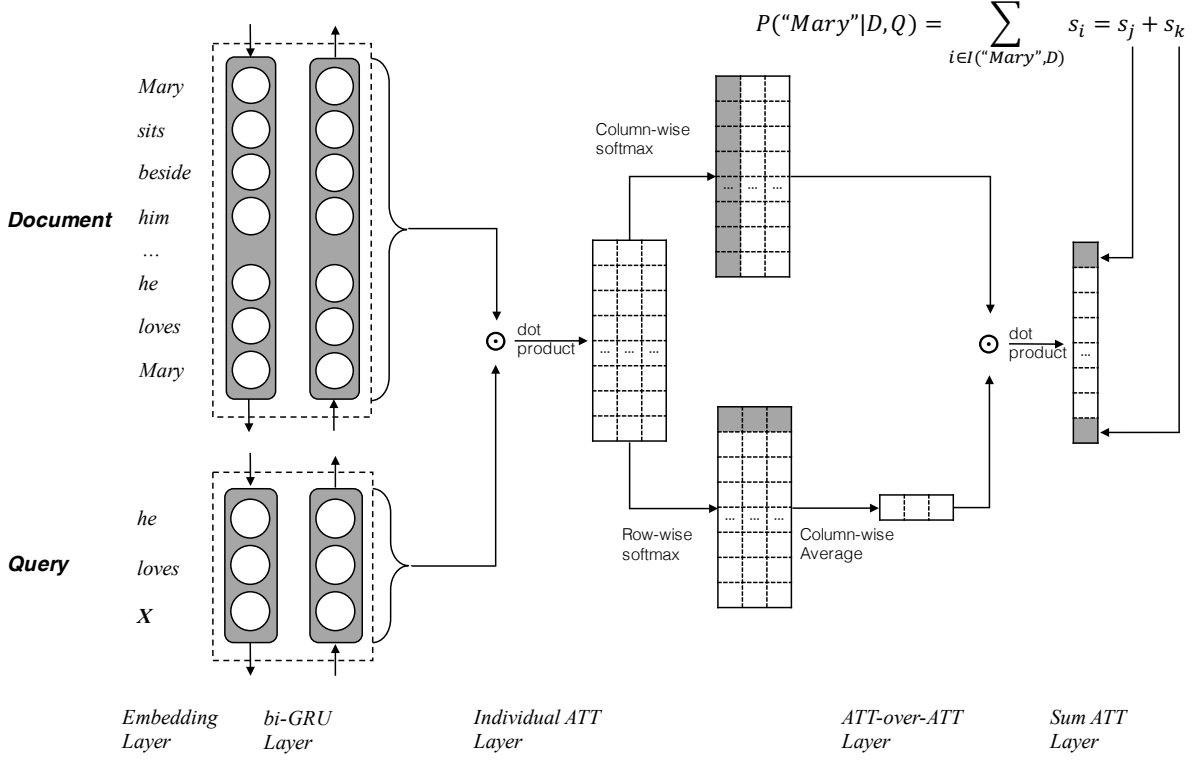[2] The CBTest datasets are available at `http://www.thespermwhale.com/jaseweston/babi/CBTest.tgz`

Figure 1: Neural network architecture of the proposed Attention-over-Attention Reader (AoA Reader).

$$e(x) = W_e * x, \ where \ x \in \mathcal{D}, \mathcal{Q} \tag{1}$$

$$\overrightarrow{h_s(x)} = \overrightarrow{GRU}(e(x)) \ ; \ \overleftarrow{h_s(x)} = \overleftarrow{GRU}(e(x)) \tag{2}$$

$$h_s(x) = [\overrightarrow{h_s(x)}; \overleftarrow{h_s(x)}] \tag{3}$$

We take $h_{doc} \in \mathbb{R}^{|\mathcal{D}|*d}$ and $h_{query} \in \mathbb{R}^{|\mathcal{Q}|*d}$ to denote the contextual representations of document and query, where d is the dimension of GRUs.

Pair-wise Matching Score. After obtaining the contextual embeddings of the document h_{doc} and query h_{query}, we calculate a pair-wise matching matrix, which indicate the pair-wise matching degree of one document word and one query word. Formally, when given ith word of document and jth word of query, we can compute a matching score by their dot product.

$$M(i,j) = h_{doc}(i) \odot h_{query}(j) \tag{4}$$

In this way, we can calculate every pair-wise matching score between the document and query, forming a matrix $M \in \mathbb{R}^{|\mathcal{D}|*|\mathcal{Q}|}$, where the value of ith row and jth column is filled by $M(i,j)$.

Individual Attentions. After getting the pair-wise matching matrix M, we apply a column-wise softmax function to get probability distributions in each column, where each column is an individual document-level attention when considering a single query word. We denote $\alpha(t) \in \mathbb{R}^{|\mathcal{D}|}$ as the document-level attention regarding query word at time t, which can be seen as a *query-to-document* attention.

$$\alpha(t) = softmax(M(1,t), ..., M(|\mathcal{D}|,t)) \tag{5}$$

$$\alpha = [\alpha(1), \alpha(2), ..., \alpha(|\mathcal{Q}|)] \tag{6}$$

Attention-over-Attention. Different from Cui et al. (2016), we use a more "wise" way to combine these document-level attentions into a final attention, while the previous work used naive heuristics, such as *summing* or *averaging* over individual attention $\alpha(t)$.

First, we calculate a reversed attention, that is, for every document word at time t, we calculate the "importance" distribution on the query, to indicate which query words are more important given a single document word. We apply a row-wise softmax function to the pair-wise matching matrix M to get query-level attentions. We denote $\beta(t) \in \mathbb{R}^{|\mathcal{Q}|}$ as the query-level attention regarding document word at time t, which can be seen as a *document-to-query* attention.

$$\beta(t) = softmax(M(t, 1), ..., M(t, |\mathcal{Q}|)) \tag{7}$$

So far, we have obtained both *query-to-document* attention α and *document-to-query* attention β. Our motivation is to exploit mutual informations between the document and query. However, most of the previous works are only relying on *query-to-document* attention, that is, only calculate one document-level attention when considering the whole query.

Then we average all the $\beta(t)$ to get an averaged query-level attention β. Note that, we do not apply another softmax to the β, because averaging individual attentions do not break the normalizing condition.

$$\beta = \frac{1}{n} \sum_{t=1}^{|\mathcal{D}|} \beta(t) \tag{8}$$

Finally, we calculate dot product of α and β to get the "attended document-level attention" $s \in \mathbb{R}^{|\mathcal{D}|}$. Intuitively, this operation is calculating a weighted sum of each individual document-level attention $\alpha(t)$ when looking at query word at time t. In this way, the contributions by each query word can be learned explicitly, and the final decision (document-level attention) is made through the voted result by the importance of each query word.

$$s = \alpha \odot \beta \tag{9}$$

Final Predictions. Following Kadlec et al. (2016), we use sum attention to get aggregated results. Note that, the final output should reflected in the vocabulary space V, rather than document-level attention $|\mathcal{D}|$, which will make a big difference in the performance, though Kadlec et al. (2016) did not illustrate this clearly.

$$P(w|\mathcal{D}, \mathcal{Q}) = \sum_{i \in I(w, \mathcal{D})} s_i, \ \ w \in V \tag{10}$$

where $I(w, \mathcal{D})$ indicate the positions that word w appear in the document $\mathcal{D}$. As the training objectives, we seek to maximize the log-likelihood of the correct answer.

$$\mathcal{L} = \sum_i \log(p(x)) \ \ , x \in \mathcal{A} \tag{11}$$

The proposed neural network architecture is depicted in Figure 1. Note that, as our model mainly adds limited steps of calculations to the AS Reader (Kadlec et al., 2016) and do not employ any additional weights, the computational complexity is similar to the AS Reader.

4 Experiments

4.1 Experimental Setups

The general settings of our neural network model are detailed below.

- Embedding Layer: The embedding weights are randomly initialized with the uniformed distribution in the interval $[-0.05, 0.05]$. While we were implementing the AS Reader, we observed that it is easy to overfit the training data within two epochs, where a similar conclusion was also made in Trischler et al. (2016). For regularization purpose and handling overfitting problems, we adopted l_2-regularization to 0.0001 and dropout rate of 0.1 (Srivastava et al., 2014). Also, it should be noted that we do not exploit any pre-trained models.

	Embed. # units	Hidden # units
CNN News	384	256
CBTest NE	384	384
CBTest CN	384	256

Table 1: Embedding and hidden layer dimensions for each task.

	CNN News			CBT NE			CBT CN		
	Train	Valid	Test	Train	Valid	Test	Train	Valid	Test
# Query	380,298	3,924	3,198	108,719	2,000	2,500	120,769	2,000	2,500
Max # candidates	527	187	396	10	10	10	10	10	10
Avg # candidates	26	26	25	10	10	10	10	10	10
Avg # tokens	762	763	716	433	412	424	470	448	461
Vocabulary	118,497			53,063			53,185		

Table 2: Statistics of public Cloze-style reading comprehension datasets: CNN news data (Hermann et al., 2015) and CBTest NE(Named Entites) / CN(Common Nouns) (Hill et al., 2015).

- Hidden Layer: Internal weights of GRUs are initialized with random orthogonal matrices (Saxe et al., 2013).

- Optimization: In order to minimize the hyper-parameter tuning, we adopted ADAM optimizer for weight updating (Kingma and Ba, 2014), with an initial learning rate of 0.001. As the GRU units still suffer from the gradient exploding issues, we set the gradient clipping threshold to 5 (Pascanu et al., 2013). We used batched training strategy of 32 samples.

Dimensions of embedding and hidden layer for each task are listed in Table 1. Due to the time limitations, we only tested a few combinations of hyper-parameters, while we expect to have a full parameter tuning in the future. The results are reported with the best model, which is selected by the performance of validation set. Implementation is done with Theano (Theano Development Team, 2016) and Keras (Chollet, 2015), and all models are trained on Tesla K40 GPU.

4.2 Results

Our experiments are carried out on public datasets: CNN news datasets (Hermann et al., 2015) and CBTest NE/CN datasets (Hill et al., 2015). The statistics of these datasets are listed in Table 2. Note that, no special treatment was applied to these datasets. Due to the time limitations, we were not able to evaluate our model in the ensemble, and only tested single model performance. We also added pretty new works to compare with (Dhingra et al., 2016; Sordoni et al., 2016; Trischler et al., 2016), which came to our attention when we are writing this paper. The experimental results are given in Table 3.
CNN News. The results on CNN news datasets shows that our AoA Reader gives competitive results among various state-of-the-art baselines, including those cutting-edge systems. When compared to the previous best result (EpiReader), our model gives similar results, where 0.4% improvements in test set and gives slight drop in validation set by 0.3%. Except for EpiReader, our model shows an absolute advantage over all other models by a significant margin.

To investigate the effectiveness of employing attention-over-attention mechanism, we also compared our model to CAS Reader, where the latter used pre-defined heuristics. As we can see that getting rid of those heuristics and letting the model to explicitly learn the weights between individual attentions could give a significant boost, where 4.1% and 3.7% improvements can be made in validation and test set.

Also, the Stanford AR (Chen et al., 2016) and GA Reader (Dhingra et al., 2016) utilized pre-trained word embeddings for initialization, while our model does not adopt any pre-trained model for initialization. Furthermore, we do not optimize for a certain type of dataset, unlike the Stanford AR only

	CNN News		CBTest NE		CBTest CN	
	Valid	Test	Valid	Test	Valid	Test
Deep LSTM Reader[1]	55.0	57.0	-	-	-	-
Attentive Reader[1]	61.6	63.0	-	-	-	-
Impatient Reader[1]	61.8	63.8	-	-	-	-
Human (context+query)[2]	-	-	-	*81.6*	-	*81.6*
LSTMs (context+query)[2]	-	-	51.2	41.8	62.6	56.0
MemNN (window + self-sup.)[2]	63.4	66.8	70.4	66.6	64.2	63.0
AS Reader[3]	68.6	69.5	73.8	68.6	68.8	63.4
CAS Reader[4]	68.2	70.0	74.2	69.2	68.2	65.7
Stanford AR[5]	72.4	72.4	-	-	-	-
*GA Reader[6]	73.0	73.8	74.9	69.0	69.0	63.9
*EpiReader[7]	**73.4**	74.0	75.3	69.7	71.5	67.4
*Iterative Attention[8]	72.6	73.3	75.2	68.6	72.1	69.2
MemNN (window + self-sup. + ensemble)[2]	66.2	69.4	-	-	-	-
AS Reader (ensemble)[3]	73.9	75.4	74.5	70.6	71.1	68.9
*GA Reader (ensemble)[6]	76.4	77.4	75.5	71.9	72.1	69.4
*EpiReader (ensemble)[7]	-	-	76.6	71.8	73.6	70.6
*Iterative Attention (ensemble)[8]	74.5	75.7	76.9	72.0	74.1	71.0
AoA Reader (single model)	73.1	**74.4**	**77.8**	**72.0**	**72.2**	**69.4**

Table 3: Results on the CNN news, CBTest NE (named entity) and CN (common noun) datasets. Results marked with [1] are taken from (Hermann et al., 2015), and [2] are taken from (Hill et al., 2015), and [3] are taken from (Kadlec et al., 2016), and [4] are taken from (Cui et al., 2016), and [5] are taken from (Chen et al., 2016), and [6] are taken from (Dhingra et al., 2016), and [7] are taken from (Trischler et al., 2016), and [8] are taken from (Sordoni et al., 2016). The result that performs best is depicted in bold face (except for ensemble models). The most recent works (within few weeks) are marked with asterisk(*).

normalized the probabilities over the named entities, rather than all the words in the document, which demonstrate that our model is more general and powerful than previous works. We have also noticed that it is fairly hard for a single model to reach above 75%, as indicated in Chen et al. (2016)'s study showed that the coreference errors (roughly takes up 25%) make the questions "unanswerable" even for the humans.

CBTest NE/CN. In CBTest NE dataset, our AoA Reader outperforms all the state-of-the-art systems by a large margin, where a 2.5% and 2.3% absolute accuracy improvements over the most recent state-of-the-art system EpiReader in the validation and test set respectively. We have also noticed that, though we haven't tried our model in ensemble, our single model could stay on par with the previous best system in the ensemble, and even we have an absolute improvement of 0.9% beyond the best ensemble model (Iterative Attention) in the validation set. This demonstrates that our model is powerful enough to compete with the ensemble models of previous works, and introducing ensembles in our model may have another boost in the performance, though we haven't tried in this paper.

In CBTest CN dataset, our model gives modest improvements over the state-of-the-art systems. When compared with Iterative Attention model, our model shows a similar result, with slight improvements on validation and test set. But when compared to EpiReader, our model could give a significant improvement with 0.3% and 1.7% gains respectively, and even larger gains over GA Reader and all of the previous works, which also demonstrate the effectiveness of our model.

5 Related Work

Cloze-style reading comprehension tasks have been widely investigated in recent studies. We will take a brief revisit to the previous works.

Hermann et al. (2015) have proposed a methodology for obtaining large quantities of $\langle \mathcal{D}, \mathcal{Q}, \mathcal{A} \rangle$ triples through news articles and its summary. Along with the release of Cloze-style reading comprehension dataset, they also proposed an attention-based neural network to tackle the issues above. Experimental results showed that the proposed neural network is effective than traditional baselines.

Hill et al. (2015) released another dataset, which stems from the children's books. Different from Hermann et al. (2015)'s work, the document and query are all generated from the raw story without any summary, which is much more general than previous work. To handle the reading comprehension task, they proposed a window-based memory network, and self-supervision heuristics is also applied to learn hard-attention.

Unlike previous works, that using blended representations of document and query to estimate the answer, Kadlec et al. (2016) proposed a simple model that directly pick the answer from the document, which is motivated by the Pointer Network (Vinyals et al., 2015). A restriction of this model is that, the answer should be a single word and appear in the document. Results on various public datasets showed that the proposed model is effective than previous works.

Apart from these progress, Liu et al. (2016) proposed to exploit these reading comprehension models into specific task. They first applied the reading comprehension model into Chinese zero pronoun resolution task with automatically generated large-scale pseudo training data. To better adapt to the zero pronoun resolution task, a two-step training procedure is also proposed, that a pre-training step and an adaptation step. The experimental results on OntoNotes 5.0 corpus showed that their method significantly outperforms various state-of-the-art systems by a large margin.

We also have noticed two very recent works (Sordoni et al., 2016) and (Trischler et al., 2016), during our writing of this article. Sordoni et al. (2016) have proposed an iterative alternating attention mechanism and gating strategies to accumulatively optimize the attention after several hops, where the number of hops is defined heuristically. Trischler et al. (2016) adopted a re-ranking strategy into the neural networks and used a joint-training method to optimize the neural network. The final prediction is determined by both the Extractor (Attention Sum Reader) and the Re-ranker (Reasoner). The works mentioned above, both outperformed the state-of-the-art systems.

Our work is primarily inspired by Cui et al. (2016) and Kadlec et al. (2016) , where the latter model is widely applied to many follow-up works (Sordoni et al., 2016; Trischler et al., 2016; Cui et al., 2016). Unlike the CAS Reader proposed by Cui et al. (2016), we do not assume any heuristics to our model, such as using merge functions: sum, avg etc. We used a mechanism called "attention-over-attention" to explicitly calculate the weights between different individual document-level attentions, and get the final attention by computing the weighted sum of them. Also, we find that our model is typically general and simple than the recently proposed model, and our model brings significant improvements over these cutting edge systems.

6 Conclusion

We present a novel neural architecture, call attention-over-attention reader, to tackle the Cloze-style reading comprehensions. The proposed Attention-over-Attention model aims to compute the attentions not only for the document but also the query side, which will benefit from the mutual information. Then a weighted sum of attention is carried out to get an attended attention over the document for the final predictions. Among several public datasets, our model could give consistent and significant improvements over various state-of-the-art systems by a large margin. A highlight in our model is that we propose to use the attention-over-attention mechanism to get "attended attention". Besides this, our model is elegant and easy to carry out but shows promising results on this task.

The future work will be carried out in the following aspects. We believe that our model is general and may apply to other tasks as well, so firstly we are going to fully investigate the usage of this architecture in other tasks. Also, we are interested to see that if the machine really "comprehend" our language by utilizing neural networks approaches, but not only serve as a "document-level" language model. In this context, we are planning to investigate the problems that need comprehensive reasoning over several sentences.

References

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

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Association for Computational Linguistics (ACL)*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

François Chollet. 2015. Keras. `https://github.com/fchollet/keras`.

Yiming Cui, Ting Liu, Zhipeng Chen, Shijin Wang, and Guoping Hu. 2016. Consensus attention-based neural networks for chinese reading comprehension. *arXiv preprint arXiv:1607.02250*.

Bhuwan Dhingra, Hanxiao Liu, William W Cohen, and Ruslan Salakhutdinov. 2016. Gated-attention readers for text comprehension. *arXiv preprint arXiv:1606.01549*.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1684–1692.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2015. The goldilocks principle: Reading children's books with explicit memory representations. *arXiv preprint arXiv:1511.02301*.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. *arXiv preprint arXiv:1603.01547*.

Diederik Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Ting Liu, Yiming Cui, Qingyu Yin, Shijin Wang, Weinan Zhang, and Guoping Hu. 2016. Generating and exploiting large-scale pseudo training data for zero pronoun resolution. *arXiv preprint arXiv:1606.01603*.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. *ICML (3)*, 28:1310–1318.

Andrew M Saxe, James L McClelland, and Surya Ganguli. 2013. Exact solutions to the nonlinear dynamics of learning in deep linear neural networks. *arXiv preprint arXiv:1312.6120*.

Alessandro Sordoni, Phillip Bachman, and Yoshua Bengio. 2016. Iterative alternating neural attention for machine reading. *arXiv preprint arXiv:1606.02245*.

Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958.

Wilson L Taylor. 1953. Cloze procedure: a new tool for measuring readability. *Journalism and Mass Communication Quarterly*, 30(4):415.

Theano Development Team. 2016. Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688, May.

Adam Trischler, Zheng Ye, Xingdi Yuan, and Kaheer Suleman. 2016. Natural language comprehension with the epireader. *arXiv preprint arXiv:1606.02270*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700.