Read and Comprehend by Gated-Attention Reader with More Belief

Haohui Deng*
Department of Computer Science
ETH Zurich
8092 Zurich, Switzerland
hadeng@student.ethz.ch

Yik-Cheung Tam*
WeChat AI
Tencent
200234 Shanghai, China
wilsontam@tencent.com

Abstract

Gated-Attention (GA) Reader has been effective for reading comprehension. GA Reader makes two assumptions: (1) a uni-directional attention that uses an input query to gate token encodings of a document; (2) encoding at the cloze position of an input query is considered for answer prediction. In this paper, we propose Collaborative Gating (CG) and Self-Belief Aggregation (SBA) to address the above assumptions respectively. In CG, we first use an input document to gate token encodings of an input query so that the influence of irrelevant query tokens may be reduced. Then the filtered query is used to gate token encodings of an document in a collaborative fashion. In SBA, we conjecture that query tokens other than the cloze token may be informative for answer prediction. We apply self-attention to link the cloze token with other tokens in a query so that the importance of query tokens with respect to the cloze position are weighted. Then their evidences are weighted, propagated and aggregated for better reading comprehension. Experiments show that our approaches advance the state-of-the-art results in CNN, Daily Mail, and Who Did What public test sets.

1 Introduction

Recently, machine reading has received a lot of attention in the research community. Several large-scale datasets of cloze-style query-document pairs have been introduced to measure machine reading capability. Deep leaning has been used for text comprehension with state-of-the-art approaches using attention mechanism. One simple and effective approach is based on Gated Attention (GA) (Dhingra et al., 2017). Viewing the attention mechanism as word alignment, GA uses document-to-query attention to align each word position of a document with a word token in a query in a "soft" manner. Then the expected encoding of the query, which can be viewed as a masking vector, is computed for each word position of a document. Through a gating function such as the element-wise product, each dimension of a token encoding in a document is interacted with the query for information filtering. Intuitively, each token of a document becomes query-aware. Through the gating mechanism, only relevant information in the document is kept for further processing. Moreover, multi-hop reasoning is applied that performs layer-wise information filtering to improve machine reading performance.

In this paper, we propose Collaborative Gating (CG) that attempts to model bi-directional information filtering between query-document pairs. We first apply query-to-document attention so that each token encoding of a query becomes *document-aware*. Then we use the filtered query and apply usual document-to-query attention to filter the document. Bi-directional attention mechanisms are performed in a collaborative manner. Multi-hop reasoning is then applied like in the GA Reader. Intuitively, bi-directional attention may capture complementary information for better machine comprehension (Seo et al., 2017; Cui et al., 2017). By filtering query-document pairs, we hope that feature representation at the final layer will be more precise for answer prediction. Our experiments have shown that CG can yield further improvement compared to GA Reader.

Another contribution is the introduction of self-attention mechanism in GA Reader. One assumption made by GA Reader is that at the final layer for answer prediction, only the cloze position of a query is considered for computing the evidence scores of entity candidates. We conjecture that surrounding words in a query may be related to the cloze position and thus provide addition-

* indicates equal contribution

Proceedings of NAACL-HLT 2018: Student Research Workshop, pages 83–91
New Orleans, Louisiana, June 2 - 4, 2018.

al evidence for answer prediction. Therefore, we employ self-attention to weight each token of the query with respect to the cloze token. Our proposed Self-Belief Aggregation (SBA) amounts to compute the expected encoding at the cloze position which can be viewed as evidence propagation from other word positions. Then similarity scores between the expected cloze token and the candidate entities of the document are computed and aggregated at the final layer. Our experiments have shown that SBA can improve machine reading performance over GA Reader.

This paper is organized as follows: In Section 2, we briefly describe related work. Section 3 gives our proposed approaches to improve GA Reader. We present experimental results in Section 4. In Section 5, we summarize and conclude with future work.

2 Related Work

The cloze-style reading comprehension task can be formulated as: Given a document-query pair (d, q), select $c \in C$ that answers the cloze position in q where C is the candidate set. Each candidate answer c appears at least once in the document d. Below are related approaches to address reading comprehension problem.

Hermann et al. (2015) employed Attentive Reader that computes a document vector via attention using q, giving a joint representation $g(d(q), q)$. In some sense, $d(q)$ becomes a query-aware representation of a document. Impatient Reader was proposed in the same paper to model the joint representation but in a incremental fashion. Stanford Reader (Chen et al., 2016) further simplified Attentive Reader with shallower recurrent units and a bilinear attention. Attention-Sum (AS) Reader introduced a bias towards frequently occurred entity candidates via summation of the probabilities of the same entity instances in a document (Kadlec et al., 2016). Cui et al. (2017) proposed Attention-over-Attention (AoA) Reader that employed a two-way attention for reading comprehension. Multi-hop architecture for text comprehension was also investigated in (Hill et al., 2016; Sordoni et al., 2016; Shen et al., 2017; Munkhdalai and Yu, 2017; Dhingra et al., 2017). Kobayashi et al. (2016) and Trischler et al. (2016) built dynamic representations for candidate answers while reading the document, sharing the same spirit to GA Reader (Dhingra et al., 2017) where token encodings of a document become query-aware. Brarda et al. (2017) proposed sequential attention to make the alignment of query and document tokens context-aware. Wang et al. (2017a) showed that additional linguistic features improve reading comprehension.

Self-attention has been successfully applied in various NLP applications including neural machine translation (Vaswani et al., 2017), abstractive summarization (Paulus et al., 2017) and sentence embedding (Lin et al., 2017). Self-attention links different positions of a sequence to generate a structural representation for the sequence. In reading comprehension literature, self-attention has been investigated. (Wang et al., 2017b) proposed a Gated Self-Matching mechanism which produced context-enhanced token encodings in a document. In this paper, we have a different angle for applying self-attention. We employ self-attention to weight and propagate evidences from different positions of a query to the cloze position to enhance reading comprehension performance.

3 Proposed Approaches

To enhance the performance of GA Reader, we propose: (1) Collaborative Gating and (2) Self-Belief Aggregation described in Section 3.1 and Section 3.2 respectively. The notations are consistent to which in original GA Reader paper (see Appendix A).

3.1 Collaborative Gating

In GA Reader, document-to-query attention is applied to obtain query-aware token encodings of a document. The attention flow is thus uni-directional. Seo et al. (2017) and Cui et al. (2017) showed that bi-directional attention can be helpful for reading comprehension. Inspired by their idea, we propose a Collaborative Gating (CG) approach under GA Reader, where *query-to-document* and *document-to-query* attention are applied in a collaborative manner. We first use query-to-document attention to generate *document-aware* query token encodings. Intuitively, we use the document to create a mask for each query token. In this step, the query is said to be "filtered" by the document. Then we use the filtered query to gate document tokens like in GA Reader. The document is said to be "filtered" by the filtered query in the previous step. The output document token encodings are fed into the nex-

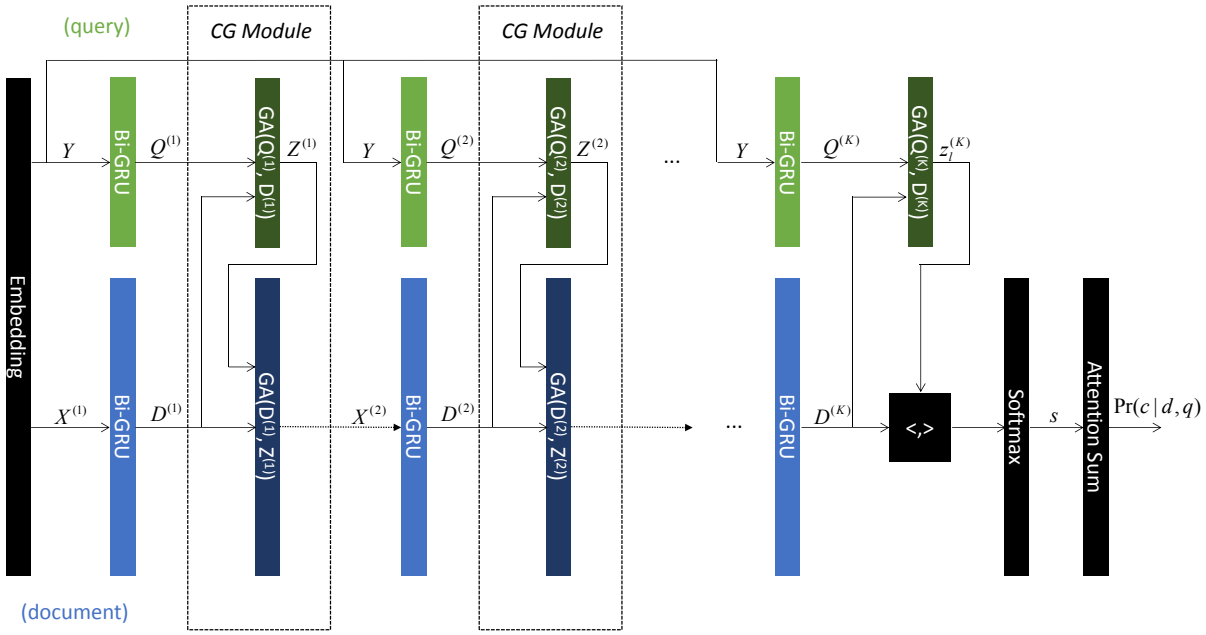

Figure 1: Collaborative Gating with a multi-hop architecture.

t computation layer. Figure 1 illustrates CG under a multi-hop architecture, showing that CG fits naturally into GA Reader. The mathematical notations are consistent to GA Reader described in Appendix A. Dashed lines represent dropout connections. CG modules are circled. At each layer, document tokens X and query tokens Y are fed into Bi-GRUs to obtain token encodings Q and D. Then we apply query-to-document attention to obtain a document-aware query representation using $GA(Q, D)$:

$$\beta_j = softmax(D^T q_j) \quad (1)$$
$$\tilde{d}_j = D\beta_j \quad (2)$$
$$z_j = q_j \odot \tilde{d}_j \quad (3)$$

Upon this, we get the filtered query tokens $Z = [z_1, z_2, ..., z_{|Q|}]$. Then we apply document-to-query attention using Z to obtain a query-aware document representation using $GA(D, Z)$:

$$\alpha_i = softmax(Z^T d_i) \quad (4)$$
$$\tilde{z}_i = Z\alpha_i \quad (5)$$
$$x_i = d_i \odot \tilde{z}_i \quad (6)$$

The resulting sequence $X = [x_1, x_2, ..., x_{|D|}]$ are fed into the next layer. We also explore another way to compute the term $\tilde{z}$ in equation 5. In particular, we may replace Z by Q in equation 5 since Q is in the unmodified encoding space compared to Z. We will study this effect in detail in Section 4.

At the final layer of GA Reader, encoding at the cloze position is used to calculate similarity score for each word token in a document. We evaluate whether applying the query-to-document attention to filter the query is crucial before computing the similarity scores. In other words, we use $D^{(K)}$ to filter the query producing $Z^{(K)}$. Then the score vector of document positions s is calculated as:

$$s = softmax((z_l^{(K)})^T D^{(K)}) \quad (7)$$

where index l is the cloze position. Similar to GA Reader, the prediction then can be obtained using equation 19 and equation 20 in Appendix A. We will study the effect of this final filtering in detail in Section 4.

3.2 Self Belief Aggregation

In this section, we introduce self-attention for GA Reader to aggregate beliefs from positions other than the cloze position. The motivation is that surrounding words other than the cloze position of a query may be informative so that beliefs from the surrounding positions can be propagated into the cloze position in a weighted manner. We employ self-attention to measure the weight between the cloze and surrounding positions. Figure 2 shows

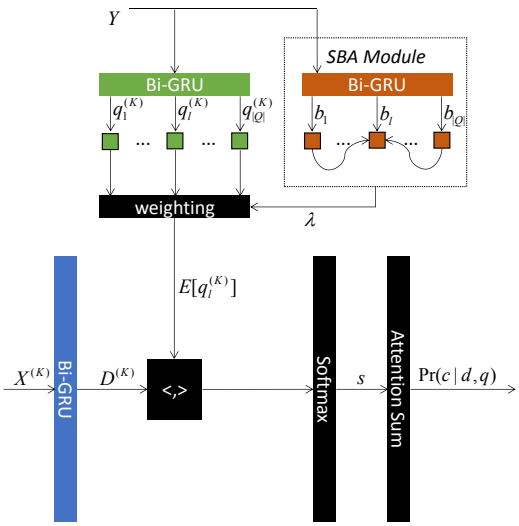

Figure 2: Self Belief Aggregation.

the Self-Belief Aggregation module at the final layer of GA Reader. Query Y is fed into the SBA module that uses another Bi-directional GRU to obtain token encodings $B = [b_1, b_2, ..., b_{|Q|}]$. Then attention weights are computed using:

$$B = \overleftrightarrow{GRU}(Y) \quad (8)$$

$$\lambda = softmax(B^T b_l) \quad (9)$$

where l is the cloze position. λ measures the importance of each query word with respect to the cloze position. We compute weighted-sum $E[q_l^{(K)}]$ using λ so that beliefs from surrounding words can be propagated and aggregated upon similarity score computation. Finally, scores at word positions of a document are calculated using:

$$s = softmax((E[q_l^{(K)}])^T D^{(K)}) \quad (10)$$

When CG is applied jointly with SBA, the filtered query $Z^{(K)}$ is used instead of $Q^{(K)}$. Namely, $E[q_l^{(K)}])$ is replaced by $E[z_l^{(K)}])$ in equation 10.

Note that self-attention can also be applied on documents to model correlation among words in documents. Considering a document sentence *"Those efforts helped him earn the 2013 CNN Hero of the Year"* and query *"@placeholder was the 2013 CNN Hero of the Year"*. Obviously, the entity co-referenced by *him* is the answer. So we hope that self-attention may have the co-reference resolution effect for "him". We will provide empirical results in Section 4.

4 Experiments

We provide experimental evaluation on our proposed approaches on public datasets in this section.

4.1 Datasets

News stories from CNN and Daily Mail (Hermann et al., 2015)[1] were used to evaluate our approaches. In particular, a query was generated by replacing an entity in the summary with @placeholder. Furthermore, entities in the news articles were anonymized to erase the world knowledge and co-occurrence effect for reading comprehension. Word embeddings of these anonymized entities are thus less informative.

Another dataset was Who Did What[2] (WDW) (Onishi et al., 2016), constructed from the LDC English Gigaword newswire corpus. Document pairs appeared around the same time period and with shared entities were chosen. Then, one article was selected as document and another article formed a cloze-style query. Queries that were answered easily by the baseline were removed to make the task more challenging. Two versions of the WDW datasets were considered for experiments: a smaller "strict" version and a larger but noisy "relaxed" version. Both shared the same validation and test sets.

4.2 Collaborative Gating Results

We evaluated Collaborative Gating under various settings. Recall from Section 3.1, we proposed two schemes for calculating the gates: Using Q or Z in equation 5. When using Z for computation, the semantics of the query are altered. When using the original Q, the semantics of the query are not altered. Moreover, we also investigate whether to apply query filtering at the final layer (denoted as "+final filtering" in Table 2).

Results show that CG helps compared to the baseline GA Reader. This may be due to the effect of query-to-document attention which makes the token encodings of a query more discriminable. Moreover, it is crucial to apply query filtering at the final layer. Using the original Q to compute the gates brought us the best results with an absolute gain of 0.7% compared to GA Reader on both the validation and test sets. Empirically, we found

[1] https://github.com/deepmind/rc-data

[2] https://tticnlp.github.io/who_did_what/

Model	CNN		Daily Mail		WDW Strict		WDW Relaxed	
	Val	Test	Val	Test	Val	Test	Val	Test
Deep LSTM Reader †	55.0	57.0	63.3	62.2	-	-	-	-
Attentive Reader †	61.6	63.0	70.5	69.0	-	53	-	55
Impatient Reader †	61.8	63.8	69.0	68.0	-	-	-	-
MemNets †	63.4	66.8	-	-	-	-	-	-
AS Reader †	68.6	69.5	75.0	73.9	-	57	-	59
DER Network †	71.3	72.9	-	-	-	-	-	-
Stanford AR †	73.8	73.6	77.6	76.6	-	64	-	65
Iterative AR †	72.6	73.3	-	-	-	-	-	-
EpiReader †	73.4	74.0	-	-	-	-	-	-
AoA Reader †	73.1	74.4	-	-	-	-	-	-
ReasoNet †	72.9	74.7	77.6	76.6	-	-	-	-
NSE †	-	-	-	-	66.5	66.2	67.0	66.7
BiDAF †	76.3	76.9	80.3	79.6	-	-	-	-
GA Reader †	77.9	77.9	81.5	80.9	71.6	71.2	72.6	72.6
MemNets (ensemble) †	66.2	69.4	-	-	-	-	-	-
AS Reader (ensemble) †	73.9	75.4	78.7	77.7	-	-	-	-
Stanford AR (ensemble) †	77.2	77.6	80.2	79.2	-	-	-	-
Iterative AR (ensemble) †	75.2	76.1	-	-	-	-	-	-
CG	**78.6**	78.6	81.9	**81.4**	**72.4**	71.9	73.0	72.6
SBA	78.5	**78.9**	**82.0**	81.2	71.5	71.5	72.3	71.3
CG + SBA	78.5	78.2	81.9	81.2	**72.4**	**72.0**	**73.1**	**72.8**

Table 1: Validation and test accuracies on CNN, Daily Mail and WDW. Results marked with † are previously published results.

Model	Accuracy	
	Val	Test
GA Reader	77.9	77.9
CG (by Z)	78.4	78.1
CG (by Q)	77.9	78.1
CG (by Z, +final filtering)	**78.7**	77.9
CG (by Q, +final filtering)	78.6	**78.6**

Table 2: Performance of Collaborative Gating under different settings on the CNN corpus.

Model	Accuracy	
	Val	Test
GA Reader	77.9	77.9
SBA on $Q^{(K)}$ (tanh)	77.1	77.1
SBA on $Q^{(K)}$	**78.5**	**78.9**
SBA on $D^{(K)}$	78.1	78.3
SBA on $D^{(K)}$&$Q^{(K)}$	78.1	78.2

Table 3: Performance of Self-Belief Aggregation under different settings on the CNN corpus.

that CG using Z for gate computation seems easier to overfit. Therefore, we use CG with the setting "by Q, +final filtering" for further comparison.

4.3 Self-Belief Aggregation Results

To study the effect of SBA, we disabled CG in the reported experiments of this section. Furthermore, we compare the attention functions using dot product and a feed forward neural network with $tanh()$ activation (Wang et al., 2017b). Results are shown in Table 3.

SBA yielded performance gain on all settings when the attention function was dot product. On the other hand, attention function using feed-forward neural network degraded accuracy compared to the baseline GA Reader which was surprising to us. Although SBA on $Q^{(K)}$ and $D^{(K)}$ individually yielded performance gain, combining them together did not bring further improvement. Even a slight drop in test accuracy was observed. Applying SBA on both query and document may make the training more difficult. From the empirical results, it seems that the learning process was led solely by document self-attention. In future work, we will consider a stepwise approach where the previous best model of a simplier network architecture will be used for initialization to avoid

Query: *in a video , @placeholder says he is sick of @entity3 being discriminated against in @entity5* **(Correct Answer: @entity18)**

GA Reader (Prediction: @entity4):

@entity4 , the leader of the @entity5 @entity9 (@entity9) , complains that @entity5 's membership of the @entity11 means it is powerless to stop a flow of foreign immigrants , many from impoverished @entity15 , into his " small island " nation . in a video posted on @entity20 , prince @entity18 said he was fed up with discrimination against @entity3 living in @entity5 .

Collaborative Gating (Prediction: @entity18):

@entity4 , the leader of the @entity5 @entity9 (@entity9) , complains that @entity5 's membership of the @entity11 means it is powerless to stop a flow of foreign immigrants , many from impoverished @entity15 , into his " small island " nation . in a video posted on @entity20 , prince @entity18 said he was fed up with discrimination against @entity3 living in @entity5 .

Self Belief Aggregation (Prediction: @entity18):

@entity4 , the leader of the @entity5 @entity9 (@entity9) , complains that @entity5 's membership of the @entity11 means it is powerless to stop a flow of foreign immigrants , many from impoverished @entity15 , into his " small island " nation . in a video posted on @entity20 , prince @entity18 said he was fed up with discrimination against @entity3 living in @entity5 .

Figure 3: Comparison between GA Reader and our proposed approaches. Entities with more red color receives higher softmax scores.

joint training from scratch. Self-attention over a long document may be difficult. Constraints such as locality may be imposed to restrict the number of word candidates in self-attention. We conjecture that modeling co-reference between entities and pronouns may be helpful compared to the full-blown self-attention over all word tokens in a document.

Figure 4 shows self-attention on two sample queries using a trained model. Surprisingly, the attention weight at the cloze position is almost

Query: *in a video , @placeholder says he is sick of @entity3 being discriminated against in @entity5*

Query: *@placeholder @entity0 built a vast business empire*

Figure 4: Self beliefs on each query positions with respect to @placeholder.

equal to unity. As a result, the weighted-sum of encodings at the cloze position reduces to encoding at the cloze position, that is the assumption of GA Reader. This may imply that SBA somehow contributes to better GA Reader training. Since the attention weight at the cloze position is almost unity, SBA can be removed during test. On the other hand, SBA did not work well on smaller datasets such as WDW.

4.4 Overall Results

We compare our approaches with previous published models as shown in Table 1. Note that CG and SBA are under the best settings reported in previous sections. CG+SBA denotes the combination of the best settings of our proposed approaches described in earlier sections. Overall, our approaches achieved the best validation and test accuracies on all datasets. On CNN and Daily Mail, CG or SBA performed similarly. But the combination of them did not always yield additional gain on all datasets. CG exploited information from query and document while SBA only used query. Although these two approaches are quite different, CG and SBA may not have strong complementary relationship for combination from the empirical results.

4.5 Significance Testing

We conducted McNemar's test on the best results we achieved using sclite toolkit [3]. The test showed the gains we achieved were all significant at 95% confidence level. To complete the test, we repeated the baseline GA Reader. Our repetition of GA Reader yielded almost the same accuracies reported by the original GA Reader paper.

5 Conclusion

We presented Collaborative Gating and Self-Belief Aggregation to optimize Gated-Attention Reader. Collaborative Gating employs document-to-query and query-to-document attentions in a collaborative and multi-hop manner. With gating mechanism, both document and query are filtered to achieve more fine-grained feature representation for machine reading. Self-Belief Aggregation attempts to propagate encodings of other query words into the cloze position using self-attention to relax the assumption of GA Reader. We evaluated our approaches on standard datasets and achieved state-of-the-art results compared to the previously published results. Collaborative Gating performed well on all datasets while SBA seems to work better on large datasets. The combination of Collaborative Gating and Self-Belief Aggregation did not bring significant additive improvements, which may imply that they are not complementary. We hope that self-attention mechnism may capture the effect of co-reference among words. So far, experimental results did not bring gain more than we hope for. Perhaps more constraints in self-attention should be imposed to learn a better model for future work. Another future investigation would be to apply SBA at each layer of GA Reader and further investigate better interaction with Collaborative Gating.

References

Rami Al-Rfou, Guillaume Alain, Amjad Almahairi, Christof Angermueller, Dzmitry Bahdanau, Nicolas Ballas, Frédéric Bastien, Justin Bayer, Anatoly Belikov, Alexander Belopolsky, et al. 2016. Theano: A python framework for fast computation of mathematical expressions. *arXiv preprint* .

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the International Conference on Learning Representations*.

Sebastian Brarda, Philip Yeres, and Samuel R. Bowman. 2017. Sequential attention: A context-aware alignment function for machine reading. In *ACL 2017 2nd Workshop on Representation Learning for NLP*.

Danqi Chen, Jason Bolton, and Christopher D Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Kyunghyun Cho, Bart Van Merriënboer, 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 Empirical Methods in Natural Language Processing*.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. 2017. Gated-attention readers for text comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Bhuwan Dhingra, Zhong Zhou, Dylan Fitzpatrick, Michael Muehl, and William W Cohen. 2016. Tweet2vec: Character-based distributed representations for social media. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

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 1693–1701.

Felix Hill, Antoine Bordes, Sumit Chopra, and Jason Weston. 2016. The goldilocks principle: Reading children's books with explicit memory representations. In *Proceedings of the 4th International Conference on Learning Representations*.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

Diederik Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the International Conference on Learning Representations*.

[3] http://www1.icsi.berkeley.edu/Speech/docs/sctk-1.2/sclite.htm

Sosuke Kobayashi, Ran Tian, Naoaki Okazaki, and Kentaro Inui. 2016. Dynamic entity representation with max-pooling improves machine reading. In *HLT-NAACL*. pages 850–855.

Peng Li, Wei Li, Zhengyan He, Xuguang Wang, Ying Cao, Jie Zhou, and Wei Xu. 2016. Dataset and neural recurrent sequence labeling model for open-domain factoid question answering. *arXiv preprint arXiv:1607.06275* .

Zhouhan Lin, Minwei Feng, Cicero Nogueira dos Santos, Mo Yu, Bing Xiang, Bowen Zhou, and Yoshua Bengio. 2017. A structured self-attentive sentence embedding. In *Proceedings of the International Conference on Learning Representations*.

Tsendsuren Munkhdalai and Hong Yu. 2017. Reasoning with memory augmented neural networks for language comprehension. In *Proceedings of the 5th International Conference on Learning Representations*.

Takeshi Onishi, Hai Wang, Mohit Bansal, Kevin Gimpel, and David McAllester. 2016. Who did what: A large-scale person-centered cloze dataset. In *Proceedings of the Empirical Methods in Natural Language Processing*.

Razvan Pascanu, Tomas Mikolov, and Yoshua Bengio. 2013. On the difficulty of training recurrent neural networks. In *International Conference on Machine Learning*. pages 1310–1318.

Romain Paulus, Caiming Xiong, and Richard Socher. 2017. A deep reinforced model for abstractive summarization. *arXiv preprint arXiv:1705.04304* .

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the empirical methods in natural language processing*. pages 1532–1543.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. In *Proceedings of the International Conference on Learning Representations*.

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM, pages 1047–1055.

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

Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. 2015. End-to-end memory networks. In *Advances in neural information processing systems*. pages 2440–2448.

Adam Trischler, Zheng Ye, Xingdi Yuan, and Kaheer Suleman. 2016. Natural language comprehension with the epireader. In *Proceedings of the Empirical Methods in Natural Language Processing*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*.

Hai Wang, Takeshi Onishi, Kevin Gimpel, and McAllester David. 2017a. Emergent predication structure in hidden state vectors of neural readers. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017b. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. volume 1, pages 189–198.

Jason Weston, Sumit Chopra, and Antoine Bordes. 2014. Memory networks. *arXiv preprint arXiv:1410.3916* .

A Gated-Attention Reader

Dhingra et al. (2017) proposed Gated-Attention Reader that combined two successful factors for text comprehension: *Multi-hop architecture* (Weston et al., 2014; Sukhbaatar et al., 2015) and *attention mechanism* (Bahdanau et al., 2015; Cho et al., 2014). At each layer, the Gated-Attention module applies attention to interact with each dimension of token encodings of a document, generating query-aware token encodings. The gated token encodings were then fed as inputs to the next layer. After a multi-hop representation learning, dot product was applied to measure the relevance between each word position in a document and the cloze position of a query. The score of each candidate entity token was calculated and summed like in the Attention-Sum Reader. Below are the details describing GA Reader computation.

Gated Recurrent Units (GRU) are used for text encoding. For an input sequence $X = [x_1, x_2, ..., x_T]$, the output sequence $H = [h_1, h_2, ..., h_T]$ can be computed as follows:

$$
\begin{aligned}
r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) \\
z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) \\
\tilde{h_t} &= tanh(W_h x_t + U_h(r_t \odot h_{t-1}) + b_h) \\
h_t &= (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h_t}
\end{aligned}
$$

where $\odot$ denotes the element-wise multiplication. r_t and z_t are *reset* and *update* gates respectively. A Bi-directional GRU (Bi-GRU) is used to process the sequence in both forward and backward directions. The produced output sequences $[h_1^f, h_2^f, ..., h_T^f]$ and $[h_1^b, h_2^b, ..., h_T^b]$ are concatenated as output encodings:

$$\overleftrightarrow{GRU}(X) = [h_1^f || h_T^b, ..., h_T^f || h_1^b] \quad (11)$$

Let $X^{(1)} = [x_1^{(1)}, x_2^{(1)}, ..., x_{|D|}^{(1)}]$ denote token embeddings of a document, and $Y = [y_1, y_2, ..., y_{|Q|}]$ denote token embeddings of a query. $|D|$ and $|Q|$ are the length of a document and a query respectively. The multi-hop architecture can be formulated as follows:

$$D^{(k)} = \overleftrightarrow{GRU}_D^{(k)}(X^{(k)}) \quad (12)$$

$$Q^{(k)} = \overleftrightarrow{GRU}_Q^{(k)}(Y) \quad (13)$$

$$X^{(k+1)} = GA(D^{(k)}, Q^{(k)}) \quad (14)$$

where $GA(D, Q)$ is a Gated-Attention module. Mathematically, it is defined as:

$$\alpha_i = softmax(Q^T d_i) \quad (15)$$

$$\tilde{q}_i = Q\alpha_i \quad (16)$$

$$x_i = d_i \odot \tilde{q}_i \quad (17)$$

where d_i is the i-th token in D. Let K be the index of the final layer, GA Reader predicts an answer using:

$$s = softmax((q_l^{(K)})^T D^{(K)}) \quad (18)$$

$$Pr(c|d, q) \propto \sum_{i \in \mathcal{I}(c,d)} s_i \quad (19)$$

$$c^* = argmax_{c \in \mathcal{C}} Pr(c|d, q) \quad (20)$$

where l is the cloze position, c is a candidate and $\mathcal{I}(c, d)$ is the set of positions where a token c appears in document d. c^* is the predicted answer.

B Implementation Details

We used the optimal configurations for CNN, Daily Mail and WDW datasets provided by (Dhingra et al., 2017) for our experiments. Our code was implemented based on the source code [4] using Theano (Al-Rfou et al., 2016). Character embedding (Dhingra et al., 2016) and the token-level indicator feature (Li et al., 2016) were used for WDW. For CNN and Daily Mail, GloVe vectors (Pennington et al., 2014) were used for word embedding initialization. We employed gradient clipping to stabilize GRU training (Pascanu et al., 2013). ADAM (Kingma and Ba, 2015) optimizer was used in all of our experiments.

[4] https://github.com/bdhingra/ga-reader