The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

StackReader: An RNN-Free Reading Comprehension Model

Yibo Jiang
Department of Computer Science
Columbia University
yj2460@columbia.edu

Zhou Zhao
College of Computer Science
Zhejiang University
zhaozhou@zju.edu.cn

Abstract

Machine comprehension of text is the problem to answer a query based on a given context. Many existing systems use RNN-based units for contextual modeling linked with some attention mechanisms. In this paper, however, we propose StackReader, an end-to-end neural network model, to solve this problem, without recurrent neural network (RNN) units and its variants. This simple model is based solely on attention mechanism and gated convolutional neural network. Experiments on SQuAD have shown to have relatively high accuracy with a significant decrease in training time.

Introduction

Machine comprehension, as a subfield of artificial intelligence, has been studied extensively in recent years. With the introduction of neural attention mechanism which allows the system to focus on targeted areas with interests, this field has witnessed many remarkable results in both computer vision and natural language processing areas. In particular, this paper is focused on reading comprehension, an application of machine comprehension in natural language processing, which requires the machine to answer a question based on paragraphs of information. In addition, this paper uses SQuAD (Rajpurkar et al. 2016) dataset where answers are phrases or sentences in the context paragraphs.

Thanks to recent developments in large benchmarks like SQuAD, reading comprehension has gained significant attraction lately as those benchmarks enable the training of end-to-end neural models. Due to the nature of text data, almost all reading comprehension models use RNN, LSTM or GRU in their models. But these RNN-based units would slow the training time because they are hard to parallelize on GPU compared to other basic operations like multiplication and convolution. On the other hand, even though LSTM or GRU can theoretically learn long distance dependences, in reality, it often only attends to local information. Because of these two limitations and inspired by recent achievements in machine translation like transformer (Vaswani et al. 2017) and ConvS2S (Gehring et al. 2017), we propose a new reading comprehension model, StackReader, that utilizes attention and gated CNN to overcome the dependence of RNN and its variants in the reading comprehension problem. To the best of our knowledge, this is the first attempt to use an RNN-free neural model to solve the reading comprehension problem.

Model Structure

The system overview is shown in Figure 1. The whole model can be roughly divided into three parts: the embedding layer, the modeling and attention layer, and the output layer.

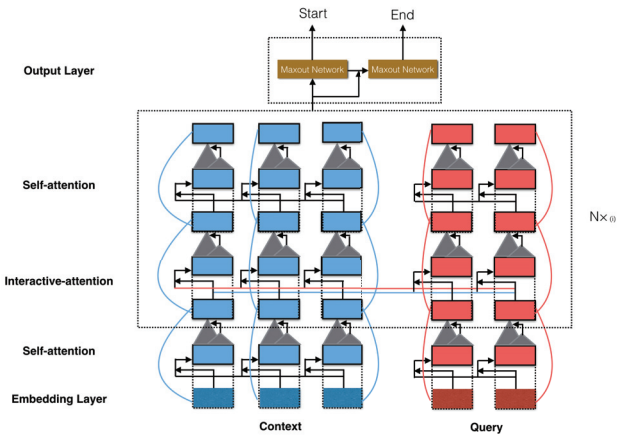

Figure 1: System Overview

Embedding Layer Let $\{c_1, c_2, ..., c_n\}$ and $\{q_1, q_2, .., q_m\}$ be words in the context paragraph and query. The first layer converts those words to the concatenation of word-level embeddings and char-level embeddings. Word-level embeddings can be acquired through pre-trained GloVe (Pennington, Socher, and Manning 2014) while we obtain char-level embeddings using Convolutional Neural Networks (Kim 2014).

Modeling and Attention Layer Unlike other popular reading comprehension models that have separate contextual modeling and attention layers. Our model tries to combine these layers in an intertwined fashion. It starts with a self-attention modeling layer and then it jumps from interactive attention layer to self-attention layer iteratively.

The interactive attention layer basically consists of multihead attention and gated linear unit (GLU). The multihead attention follows from Google's transformer (Vaswani et al. 2017). In particular, let $\mathbf{C}_t$ and $\mathbf{Q}_t$ be the representations of the context and the query at a given time frame t. Then,

$$\hat{\mathbf{C}}_t = Maxout(MultiHeadAttend(\mathbf{C}_t, \mathbf{Q}_t)) \\ = Maxout(Concat(head_1, ..., head_h))$$

Where

$$head_i = Attention(\mathbf{C}_t W_i^1, \mathbf{Q}_t W_i^2, \mathbf{C}_t W_i^3)$$

and $Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$. d_k is the dimension of the vector and h is the number of attention heads in the system. Then final output will then go through a GLU based on $\hat{\mathbf{C}}_t$ and $\mathbf{C}_t$ with residual connection.

$$\mathbf{C}_{t+1} = residual\_fn(\hat{\mathbf{C}}_{t+1}, \mathbf{C}_t)$$

$$\hat{\mathbf{C}}_{t+1} = conv(\mathbf{C}_{new}) \times sigmoid(conv(\mathbf{C}_{new}))$$

$$\mathbf{C}_{new} = [\hat{\mathbf{C}}_t; \mathbf{C}_t]$$

$conv$ is just a one-dimensional CNN. A similar operation will be performed for $\mathbf{Q}_t$ as well in the interactive layer.

The self-attention layer is similar to the interactive layer except that the attention now is based on query and context themselves.

The intuition behind using gated CNN after multihead attention is simple. Multihead attention can help solve the long dependence issue LSTM has. But it lacks the gating mechanism embedded in LSTM. So we add gated CNN to help solve this problem.

Output Layer Because the SQuAD dataset requires a phrase or sentence in the context for the answer, we only need to predict the start and end indices of the answer. We denote the start and end indices probability distributions as a_1 and a_2. Let $\mathbf{C}_T$ and $\mathbf{Q}_T$ be the last outputs from the modeling and attention layer. Then

$$a_1 = Maxout(\mathbf{C}_T)$$

For a_2, we first pass $\mathbf{C}_T$ to a GLU to get $\hat{\mathbf{C}}_T$. Then we obtain A which is a weighted sum of $\mathbf{C}_T$ based on a_1. Finally we get S based on $\mathbf{C}_T$ and $\mathbf{Q}_T$ using attention over attention model (AoA) (Cui et al. 2016). Then

$$a_2 = Maxout([\hat{\mathbf{C}}_T; A; S; \mathbf{C}_T])$$

Experiments and Results

The experiments are tested on SQuAD data set (Rajpurkar et al. 2016). In particular, the results shown here are only tested on the development data set available to the public. The number of units used in the maxout network throughout the system is 5. There are in total 7 interactive attention layers and 8 self-attention layers. There are 5 attention heads in multihead attention. We also apply 0.8 dropout rate for the residual connection. The model is trained on one Tesla K40 GPU with 12GB RAM for 24k training steps.

As shown in the Table 1, we can achieve comparable result to BiDAF (Seo et al. 2016) on developmental data. But our model is much faster. BiDAF takes around 16.5 hours to train and our model only needs around 8 hours.

Method	EM	F1	Training Time
StackReader (Ours)	65.86	75.70	$\sim$ 8hrs
BiDAF (Single)	67.7	77.3	$\sim$ 16.5hrs

Table 1: Comparison between our model and BiDAF on SQuDA's developmental data set

Conclusion

In this paper, we present a new model for reading comprehension based solely on neural attention and gated CNN. It has achieved comparable results to some popular models like BiDAF but with a significant decrease in training time. Future works include further improving the model to achieve better accuracy. Another improvement would be to transform the model into a binarized neural network to reduce the size and increase the speed of the model. Researchers have shown that binarized network can work with CNN (Courbariaux et al. 2016) but its power on attention models is still an open question.

References

Courbariaux, M.; Hubara, I.; Soudry, D.; El-Yaniv, R.; and Bengio, Y. 2016. Binarized neural networks: Training deep neural networks with weights and activations constrained to+ 1 or-1. *arXiv preprint arXiv:1602.02830*.

Cui, Y.; Chen, Z.; Wei, S.; Wang, S.; Liu, T.; and Hu, G. 2016. Attention-over-attention neural networks for reading comprehension. *arXiv preprint arXiv:1607.04423*.

Gehring, J.; Auli, M.; Grangier, D.; Yarats, D.; and Dauphin, Y. N. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, 1243–1252.

Kim, Y. 2014. Convolutional neural networks for sentence classification. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, 1746–1751.

Pennington, J.; Socher, R.; and Manning, C. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 1532–1543.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Seo, M.; Kembhavi, A.; Farhadi, A.; and Hajishirzi, H. 2016. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention is all you need. *CoRR* abs/1706.03762.