R-NET: MACHINE READING COMPREHENSION WITH SELF-MATCHING NETWORKS*

Natural Language Computing Group, Microsoft Research Asia†

ABSTRACT

In this paper, we introduce R-NET, an end-to-end neural networks model for reading comprehension style question answering, which aims to answer questions from a given passage. We first match the question and passage with gated attention-based recurrent networks to obtain the question-aware passage representation. Then we propose a self-matching attention mechanism to refine the representation by matching the passage against itself, which effectively encodes information from the whole passage. We finally employ the pointer networks to locate the positions of answers from the passages. We conduct extensive experiments on the SQuAD and MS-MARCO datasets, and our model achieves the best results on both datasets among all published results.

1 INTRODUCTION

In this paper, we focus on reading comprehension style question answering which aims to answer questions given a passage or document. We mainly focus on the Stanford Question Answering Dataset (SQuAD) (Rajpurkar et al., 2016) and Microsoft MAchine Reading COmprehension (MS-MARCO) dataset, two large-scale datasets for reading comprehension and question answering which are both manually created through crowdsourcing. SQuAD requires to answer questions given a passage. It constrains answers to the space of all possible spans within the reference passage, which is different from cloze-style reading comprehension datasets (Hermann et al., 2015; Hill et al., 2016) in which answers are single words or entities. Moreover, SQuAD requires different forms of logical reasoning to infer the answer (Rajpurkar et al., 2016). Another real dataset, MS-MARCO provides several related documents collected from Bing Index for a question. The answer to the question in MS-MARCO is generated by human and the answer words can not only come from the given text.

Rapid progress has been made since the release of the SQuAD dataset. Wang & Jiang (2016b) build question-aware passage representation with match-LSTM (Wang & Jiang, 2016a), and predict answer boundaries in the passage with pointer networks (Vinyals et al., 2015). Seo et al. (2016) introduce bi-directional attention flow networks to model question-passage pairs at multiple levels of granularity. Xiong et al. (2016) propose dynamic co-attention networks which attend the question and passage simultaneously and iteratively refine answer predictions. Lee et al. (2016) and Yu et al. (2016) predict answers by ranking continuous text spans within passages.

Inspired by Wang & Jiang (2016b), we introduce R-NET, illustrated in Figure 1, an end-to-end neural network model for reading comprehension and question answering. Our model consists of four parts: 1) the recurrent network encoder to build representation for questions and passages separately, 2) the gated matching layer to match the question and passage, 3) the self-matching layer to aggregate information from the whole passage, and 4) the pointer-network based answer boundary prediction layer. The key contributions of this work are three-fold.

* This is the work-in-progress technical report of our system and algorithm, namely R-NET, for the machine reading comprehension task. We will update this technical report when there are significant improvements of R-NET on the SQuAD leaderboard. An early version of this technical report, namely "Gated Self-Matching Networks for Reading Comprehension and Question Answering. Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang and Ming Zhou", has been accepted by and will be presented in ACL 2017.

† Please contact Furu Wei and Ming Zhou for the machine reading comprehension research in Microsoft Research Asia.

Passage: Tesla later approached Morgan to ask for more funds to build a more powerful transmitter. **When asked where all the money had gone, Tesla responded by saying that he was affected by the Panic of 1901**, which he (Morgan) had caused. Morgan was shocked by the reminder of his part in the stock market crash and by Tesla's breach of contract by asking for more funds. Tesla wrote another plea to Morgan, but it was also fruitless. Morgan still owed Tesla money on the original agreement, and Tesla had been facing foreclosure even before construction of the tower began.
Question: On what did Tesla blame for the loss of the initial money?
Answer: Panic of 1901

Table 1: An example from the SQuAD dataset.

First, we propose a gated attention-based recurrent network, which adds an additional gate to the attention-based recurrent networks (Bahdanau et al., 2014; Rocktäschel et al., 2015; Wang & Jiang, 2016a), to account for the fact that words in the passage are of different importance to answer a particular question for reading comprehension and question answering. In Wang & Jiang (2016a), words in a passage with their corresponding attention-weighted question context are encoded together to produce question-aware passage representation. By introducing a gating mechanism, our gated attention-based recurrent network assigns different levels of importance to passage parts depending on their relevance to the question, masking out irrelevant passage parts and emphasizing the important ones.

Second, we introduce a self-matching mechanism, which can effectively aggregate evidence from the whole passage to infer the answer. Through a gated matching layer, the resulting question-aware passage representation effectively encodes question information for each passage word. However, recurrent networks can only memorize limited passage context in practice despite its theoretical capability. One answer candidate is often unaware of the clues in other parts of the passage. To address this problem, we propose a self-matching layer to dynamically refine passage representation with information from the whole passage. Based on question-aware passage representation, we employ gated attention-based recurrent networks on passage against passage itself, aggregating evidence relevant to the current passage word from every word in the passage. A gated attention-based recurrent network layer and self-matching layer dynamically enrich each passage representation with information aggregated from both question and passage, enabling subsequent network to better predict answers.

Lastly, the proposed method yields state-of-the-art results against strong baselines. Our single model achieves 72.3% exact match accuracy on the hidden SQuAD test set, while the ensemble model further boosts the result to 76.9%, which currently[1] holds the first place on the SQuAD leaderboard. Besides, our model also achieves the best published results on MS-MARCO dataset (Nguyen et al., 2016).

2 Task Description

For reading comprehension style question answering, a passage **P** and question **Q** are given, our task is to predict an answer **A** to question **Q** based on information found in **P**. The SQuAD dataset further constrains answer **A** to be a continuous sub-span of passage **P**. Answer **A** often includes non-entities and can be much longer phrases. This setup challenges us to understand and reason about both the question and passage in order to infer the answer. Table 1 shows a simple example from the SQuAD dataset. As for MS-MARCO dataset, several related passages **P** from Bing Index are provided for a question **Q**. Besides, the answer **A** in MS-MARCO is generated by human which can not be a continuous sub-span of the passage.

3 R-NET Structure

Figure 1 gives an overview of R-NET. First, the question and passage are processed by a bi-directional recurrent network (Mikolov et al., 2010) separately. We then match the question and passage with gated attention-based recurrent networks, obtaining question-aware representation for

[1] On May. 6, 2017

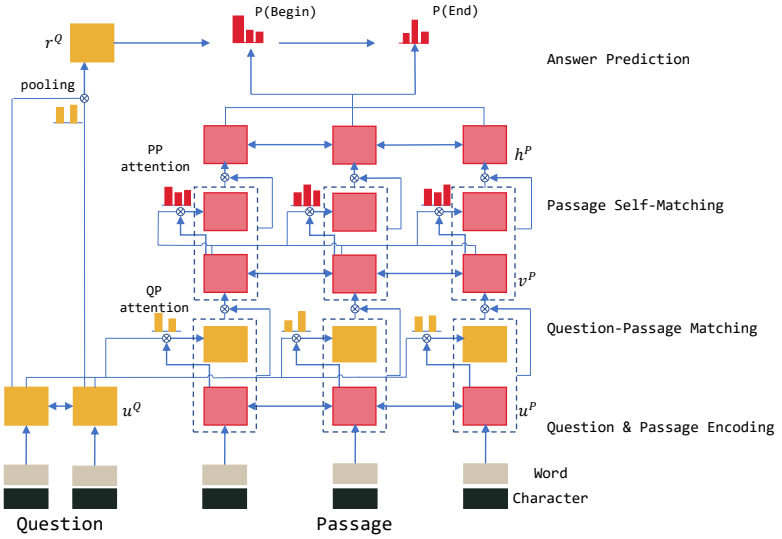

Figure 1: R-NET structure overview.

the passage. On top of that, we apply self-matching attention to aggregate evidence from the whole passage and refine the passage representation, which is then fed into the output layer to predict the boundary of the answer span.

3.1 Question and Passage Encoder

Consider a question Q = $\{w_t^Q\}_{t=1}^m$ and a passage P = $\{w_t^P\}_{t=1}^n$. We first convert the words to their respective word-level embeddings ($\{e_t^Q\}_{t=1}^m$ and $\{e_t^P\}_{t=1}^n$) and character-level embeddings ($\{c_t^Q\}_{t=1}^m$ and $\{c_t^P\}_{t=1}^n$). The character-level embeddings are generated by taking the final hidden states of a bi-directional recurrent neural network (RNN) applied to embeddings of characters in the token. Such character-level embeddings have been shown to be helpful to deal with out-of-vocab (OOV) tokens. We then use a bi-directional RNN to produce new representation $u_1^Q, \ldots, u_m^Q$ and $u_1^P, \ldots, u_n^P$ of all words in the question and passage respectively:

$$u_t^Q = \text{BiRNN}_Q(u_{t-1}^Q, [e_t^Q, c_t^Q]) \tag{1}$$

$$u_t^P = \text{BiRNN}_P(u_{t-1}^P, [e_t^P, c_t^P]) \tag{2}$$

We choose to use Gated Recurrent Unit (GRU) (Cho et al., 2014) in our experiment since it performs similarly to LSTM (Hochreiter & Schmidhuber, 1997) but is computationally cheaper.

3.2 Gated Attention-based Recurrent Networks

We propose a gated attention-based recurrent network to incorporate question information into passage representation. It is a variant of attention-based recurrent networks, with an additional gate to determine the importance of information in the passage regarding a question. Given question and passage representation $\{u_t^Q\}_{t=1}^m$ and $\{u_t^P\}_{t=1}^n$, Rocktäschel et al. (2015) propose generating sentence-pair representation $\{v_t^P\}_{t=1}^n$ via soft-alignment of words in the question and passage as follows:

$$v_t^P = \text{RNN}(v_{t-1}^P, c_t) \tag{3}$$

where $c_t = att(u^Q, [u_t^P, v_{t-1}^P])$ is an attention-pooling vector of the whole question (u^Q):

$$\begin{aligned} s_j^t &= \text{v}^\text{T}\tanh(W_u^Q u_j^Q + W_u^P u_t^P + W_v^P v_{t-1}^P) \\ a_i^t &= \exp(s_i^t)/\Sigma_{j=1}^m \exp(s_j^t) \\ c_t &= \Sigma_{i=1}^m a_i^t u_i^Q \end{aligned} \tag{4}$$

Each passage representation v_t^P dynamically incorporates aggregated matching information from the whole question.

Wang & Jiang (2016a) introduce match-LSTM, which takes u_t^P as an additional input into the recurrent network:

$$v_t^P = \text{RNN}(v_{t-1}^P, [u_t^P, c_t]) \tag{5}$$

To determine the importance of passage parts and attend to the ones relevant to the question, we add another gate to the input ($[u_t^P, c_t]$) of RNN:

$$\begin{aligned} g_t &= \text{sigmoid}(W_g[u_t^P, c_t]) \\ [u_t^P, c_t]^* &= g_t \odot [u_t^P, c_t] \end{aligned} \tag{6}$$

Different from the gates in LSTM or GRU, the additional gate is based on the current passage word and its attention-pooling vector of the question, which focuses on the relation between the question and current passage word. The gate effectively model the phenomenon that only parts of the passage are relevant to the question in reading comprehension and question answering. $[u_t^P, c_t]^*$ is utilized in subsequent calculations instead of $[u_t^P, c_t]$. We call this gated attention-based recurrent networks.

3.3 Self-Matching Attention

Through gated attention-based recurrent networks, question-aware passage representation $\{v_t^P\}_{t=1}^n$ is generated to pinpoint important parts in the passage. One problem with such representation is that it has very limited knowledge of context. One answer candidate is often oblivious to important cues in the passage outside its surrounding window. Moreover, there exists some sort of lexical or syntactic divergence between the question and passage in the majority of SQuAD dataset (Rajpurkar et al., 2016). Passage context is necessary to infer the answer. To address this problem, we propose directly matching the question-aware passage representation against itself. It dynamically collects evidence from the whole passage for words in passage and encodes the evidence relevant to the current passage word and its matching question information into the passage representation h_t^P:

$$h_t^P = \text{BiRNN}(h_{t-1}^P, [v_t^P, c_t]) \tag{7}$$

where $c_t = att(v^P, v_t^P)$ is an attention-pooling vector of the whole passage (v^P):

$$\begin{aligned} s_j^t &= \text{v}^{\text{T}}\tanh(W_v^P v_j^P + W_v^{\tilde{P}} v_t^P) \\ a_i^t &= \exp(s_i^t)/\Sigma_{j=1}^n \exp(s_j^t) \\ c_t &= \Sigma_{i=1}^n a_i^t v_i^P \end{aligned} \tag{8}$$

An additional gate as in gated attention-based recurrent networks is applied to $[v_t^P, c_t]$ to adaptively control the input of RNN.

Self-matching extracts evidence from the whole passage according to the current passage word and question information.

3.4 Output Layer

We follow Wang & Jiang (2016b) and use pointer networks (Vinyals et al., 2015) to predict the start and end position of the answer. In addition, we use an attention-pooling over the question representation to generate the initial hidden vector for the pointer network. Given the passage representation $\{h_t^P\}_{t=1}^n$, the attention mechanism is utilized as a pointer to select the start position (p^1) and end position (p^2) from the passage, which can be formulated as follows:

$$\begin{aligned} s_j^t &= \text{v}^{\text{T}}\tanh(W_h^P h_j^P + W_{\text{h}}^a h_{t-1}^a) \\ a_i^t &= \exp(s_i^t)/\Sigma_{j=1}^n \exp(s_j^t) \\ p^t &= \text{argmax}(a_1^t, \ldots, a_n^t) \end{aligned} \tag{9}$$

Here h_{t-1}^a represents the last hidden state of the answer recurrent network (pointer network). The input of the answer recurrent network is the attention-pooling vector based on current predicted probability a^t:

$$\begin{aligned} c_t &= \Sigma_{i=1}^n a_i^t h_i^P \\ h_t^a &= \text{RNN}(h_{t-1}^a, c_t) \end{aligned} \tag{10}$$

When predicting the start position, h^a_{t-1} represents the initial hidden state of the answer recurrent network. We utilize the question vector r^Q as the initial state of the answer recurrent network. $r^Q = att(u^Q, V^Q_r)$ is an attention-pooling vector of the question based on the parameter V^Q_r:

$$\begin{aligned} s_j &= \mathrm{v}^{\mathrm{T}} \tanh(W^Q_u u^Q_j + W^Q_{\mathrm{v}} V^Q_r) \\ a_i &= \exp(s_i)/\Sigma^m_{j=1}\exp(s_j) \\ r^Q &= \Sigma^m_{i=1} a_i u^Q_i \end{aligned} \tag{11}$$

To train the network, we minimize the sum of the negative log probabilities of the ground truth start and end position by the predicted distributions.

4 EXPERIMENT

4.1 IMPLEMENTATION DETAILS

We mainly focus on the SQuAD dataset to train and evaluate our model, which has garnered a huge attention over the past few months. SQuAD is composed of 100,000+ questions posed by crowd workers on 536 Wikipedia articles. The dataset is randomly partitioned into a training set (80%), a development set (10%), and a test set (10%). The answer to every question is a segment of the corresponding passage.

We use the tokenizer from Stanford CoreNLP (Manning et al., 2014) to preprocess each passage and question. The Gated Recurrent Unit (Cho et al., 2014) variant of LSTM is used throughout our model. For word embedding, we use pre-trained case-sensitive GloVe embeddings[2] (Pennington et al., 2014) for both questions and passages, and it is fixed during training; We use zero vectors to represent all out-of-vocab words. We utilize 1 layer of bi-directional GRU to compute character-level embeddings and 3 layers of bi-directional GRU to encode questions and passages, the gated attention-based recurrent network for question and passage matching is also encoded bidirectionally in our experiment. The hidden vector length is set to 75 for all layers. The hidden size used to compute attention scores is also 75. We also apply dropout (Srivastava et al., 2014) between layers with a dropout rate of 0.2. The model is optimized with AdaDelta (Zeiler, 2012) with an initial learning rate of 1. The ρ and ϵ used in AdaDelta are 0.95 and $1e^{-6}$ respectively.

4.2 MAIN RESULTS

Two metrics are utilized to evaluate model performance of SQuAD: Exact Match (EM) and F1 score. EM measures the percentage of the prediction that matches one of the ground truth answers exactly. F1 measures the overlap between the prediction and ground truth answers which takes the maximum F1 over all of the ground truth answers. The scores on dev set are evaluated by the official script[3]. Since the test set is hidden, we are required to submit the model to Stanford NLP group to obtain the test scores.

Table 2 shows exact match and F1 scores on the dev and test set of our model and competing approaches[4]. The ensemble model consists of 18 training runs with the identical architecture and hyper-parameters. At test time, we choose the answer with the highest sum of confidence scores amongst the 18 runs for each question. As we can see, our method clearly outperforms the baseline and several strong state-of-the-art systems for both single model and ensembles. **R-NET (March, 2017)** entry refers to results obtained with our improvement after ACL submission. After the original self-matching layer of the passage, we utilize bi-directional GRU to deeply integrate the matching results before feeding them into answer pointer layer. It helps to further propagate the information aggregated by self-matching of the passage.

[2]Downloaded from `http://nlp.stanford.edu/data/glove.840B.300d.zip`.

[3]Downloaded from `http://stanford-qa.com`

[4]Extracted from SQuAD leaderboard `http://stanford-qa.com` on May. 6, 2017.

	Dev Set	Test Set
Single model	**EM / F1**	**EM / F1**
LR Baseline (Rajpurkar et al., 2016)	40.0 / 51.0	40.4 / 51.0
Dynamic Chunk Reader (Yu et al., 2016)	62.5 / 71.2	62.5 / 71.0
Attentive CNN context with LSTM (NLPR, CASIA)	- / -	63.3 / 73.5
Match-LSTM with Ans-Ptr (Wang & Jiang, 2016b)	64.1 / 73.9	64.7 / 73.7
Dynamic Coattention Networks (Xiong et al., 2016)	65.4 / 75.6	66.2 / 75.9
Iterative Coattention Network (Fudan University)	- / -	67.5 / 76.8
FastQA (Weissenborn et al., 2017)	- / -	68.4 / 77.1
BiDAF (Seo et al., 2016)	68.0 / 77.3	68.0 / 77.3
T-gating (Peking University)	- / -	68.1 / 77.6
RaSoR (Lee et al., 2016)	- / -	69.6 / 77.7
SEDT+BiDAF (Liu et al., 2017)	- / -	68.5 / 78.0
Multi-Perspective Matching (Wang et al., 2016)	- / -	70.4 / 78.8
FastQAExt (Weissenborn et al., 2017)	- / -	70.8 / 78.9
Mnemonic Reader (NUDT & Fudan University)	- / -	69.9 / 79.2
Document Reader (Chen et al., 2017)	- / -	70.7 / 79.4
ReasoNet (Shen et al., 2016)	- / -	70.6 / 79.4
Ruminating Reader (Gong & Bowman, 2017)	- / -	70.6 / 79.5
jNet (Zhang et al., 2017)	- / -	70.6 / 79.8
Interactive AoA Reader (Joint Laboratory of HIT and iFLYTEK Research)	- / -	71.2 / 79.9
R-NET (Wang et al., 2017)	**71.1 / 79.5**	**71.3 / 79.7**
R-NET (March 2017)	**72.3 / 80.6**	**72.3 / 80.7**
Ensemble model		
Fine-Grained Gating (Yang et al., 2016)	62.4 / 73.4	62.5 / 73.3
Match-LSTM with Ans-Ptr (Wang & Jiang, 2016b)	67.6 / 76.8	67.9 / 77.0
QFASE (NUS)	- / -	71.9 / 80.0
Dynamic Coattention Networks (Xiong et al., 2016)	70.3 / 79.4	71.6 / 80.4
T-gating (Peking University)	- / -	72.8 / 81.0
Multi-Perspective Matching (Wang et al., 2016)	- / -	73.8 / 81.3
jNet (Zhang et al., 2017)	- / -	73.0 / 81.5
BiDAF (Seo et al., 2016)	- / -	73.7 / 81.5
SEDT+BiDAF (Liu et al., 2017)	- / -	73.7 / 81.5
Mnemonic Reader (NUDT & Fudan University)	- / -	73.7 / 81.7
ReasoNet (Shen et al., 2016)	- / -	75.0 / 82.6
R-NET (Wang et al., 2017)	**75.6 / 82.8**	**75.9 / 82.9**
R-NET (March 2017)	**76.7 / 83.7**	**76.9 / 84.0**
Human Performance (Rajpurkar et al., 2016)	- / -	82.3 / 91.2

Table 2: The performance of our R-NET and competing approaches[4] on SQuAD dataset.

Single Model	ROUGE-L / BLEU1
FastQAExt (Weissenborn et al., 2017)	33.7 / 33.9
Prediction (Wang & Jiang, 2016b)	37.3 / 40.7
ReasoNet (Shen et al., 2016)	38.8 / 39.9
R-NET	42.9 / 42.2

Table 3: The performance of our R-NET and competing approaches[5] on MS-MARCO dataset.

4.3 MS-MARCO RESULT

We also apply our method to MS-MARCO dataset (Nguyen et al., 2016). MS-MARCO is another machine comprehension dataset, with two key differences from SQuAD. In MS-MARCO, every question has several corresponding passages, so we simply concatenate all passages of one question in the order that given in the dataset. Secondly, the answers in MS-MARCO are not necessarily sub-spans of the passages so that the metrics in the official tool of MS-MARCO evaluation are BLEU and ROUGE-L, which are widely used in many domains. In this regard, we choose the span with the highest ROUGE-L score with the reference answer as the gold span in the training, and predict the highest scoring span as answer during prediction. We train our model on MS-MARCO dataset, and the results (Table 3) show that our method out-performs other competitive baselines[5].

4.4 DISCUSSIONS

In this section, we report and discuss some efforts that failed to bring improvements in our experiments. As with all empirical findings on SQuAD, results reported here only apply to our exact settings. The findings do not necessarily indicate the effectiveness of the discussed methods when used to other datasets or combined with baseline models different from ours. We believe these directions are valuable research topics and we are experimenting these ideas with different models and implementations.

1. **Sentence Ranking** In SQuAD, the passage consists of several sentences and the answer span always falls into one sentence. It is natural to consider whether ranking sentence would help locate the final answer. We have tried two ways to integrate sentence ranking information: (a) we trained a separate sentence ranking model, and combined this model with the span prediction model; (b) we treat span prediction and sentence prediction as two related task, and trained a multi-task model. Both methods failed to improve the final results. Analysis shows that the sentence models consistently under-perform the span prediction model even on sentence prediction task. Our best sentence model achieves accuracy of 86%, while our span prediction model has over 92% accuracy predicting the answer sentence. This indicates that the exact span information is in fact critical in selecting the correct answer sentence.
2. **Syntax Information** We have tried three methods to integrate syntax information into our model. Firstly, we have tried to add some syntax features as input in encoding layers. These syntax features include POS tags, NER results, linearized PCFG tree tags and dependency labels. Secondly, we have tried to integrate a tree-LSTM style module after our encoding layer. We use a multi-input LSTM to build hidden states following dependency tree paths in both top-down and bottom-up passes. Lastly, we tried to use dependency parsing as an additional task in a multi-task setting. All the above failed to bring any benefit to our model on SQuAD dataset.
3. **Multi-hop Inference** We have tried to add multi-hop inference modules in the answer pointer layer, but failed to get improvements on the final results in the context of the current R-NET network structure. One reason might be that the questions which require such inference are too complex to learn effectively under current settings, especially considering there are no annotations about explicit inference process in SQuAD.

[5]Results except ours are extracted from MS-MARCO leaderboard `http://www.msmarco.org/leaders.aspx` on May. 6, 2017.

4. **Question Generation** For data-driven approach, labeled data might become the bottleneck for better performance. While texts are abundant, it is not easy to find question-passage pairs that match the style of SQuAD. To generate more data, we trained a sequence-to-sequence question generation model using SQuAD dataset (Zhou et al., 2017), and produced a large amount of pseudo question-passage pairs from English Wikipedia. We trained a R-NET model on this pseudo corpus together with SQuAD training data, and we assigned a smaller weight to auto-generated samples so that the total weights of pseudo corpus and real corpus are about equal. So far, such approach failed to make any gains in the final results. Analysis shows that the quality of generated questions needs improvement.

5 Related Work

Reading Comprehension and Question Answering Dataset Benchmark datasets play an important role in recent progress in reading comprehension and question answering research. Existing datasets can be classified into two categories according to whether they are manually labeled. Those that are labeled by humans are always in high quality (Richardson et al., 2013; Berant et al., 2014; Yang et al., 2015), but are too small for training modern data-intensive models. Those that are automatically generated from natural occurring data can be very large (Hill et al., 2016; Hermann et al., 2015), which allow the training of more expressive models. However, they are in cloze style, in which the goal is to predict the missing word (often a named entity) in a passage. Moreover, Chen et al. (2016) have shown that the CNN / Daily News dataset (Hermann et al., 2015) requires less reasoning than previously thought, and conclude that performance is almost saturated.

Different from above datasets, the SQuAD provides a large and high-quality dataset. The answers in SQuAD often include non-entities and can be much longer phrase, which is more challenging than cloze-style datasets. Moreover, Rajpurkar et al. (2016) show that the dataset retains a diverse set of answers and requires different forms of logical reasoning, including multi-sentence reasoning. MS MARCO (Nguyen et al., 2016) is also a large-scale dataset. The questions in the dataset are real anonymized queries issued through Bing or Cortana and the passages are related web pages. For each question in the dataset, several related passages are provided. However, the answers are human generated, which is different from SQuAD where answers must be a span of the passage.

End-to-end Neural Networks for Reading Comprehension Along with cloze-style datasets, several powerful deep learning models (Hermann et al., 2015; Hill et al., 2016; Chen et al., 2016; Kadlec et al., 2016; Sordoni et al., 2016; Cui et al., 2016; Trischler et al., 2016; Dhingra et al., 2016; Shen et al., 2016) have been introduced to solve this problem. Hermann et al. (2015) first introduce attention mechanism into reading comprehension. Hill et al. (2016) propose a window-based memory network for CBT dataset. Kadlec et al. (2016) introduce pointer networks with one attention step to predict the blanking out entities. Sordoni et al. (2016) propose an iterative alternating attention mechanism to better model the links between question and passage. Trischler et al. (2016) solve cloze-style question answering task by combining an attentive model with a reranking model. Dhingra et al. (2016) propose iteratively selecting important parts of the passage by a multiplying gating function with the question representation. Cui et al. (2016) propose a two-way attention mechanism to encode the passage and question mutually. Shen et al. (2016) propose iteratively inferring the answer with a dynamic number of reasoning steps and is trained with reinforcement learning.

Neural network-based models demonstrate the effectiveness on the SQuAD dataset. Wang & Jiang (2016b) combine match-LSTM and pointer networks to produce the boundary of the answer. Xiong et al. (2016) and Seo et al. (2016) employ variant coattention mechanism to match the question and passage mutually. Xiong et al. (2016) propose a dynamic pointer network to iteratively infer the answer. Yu et al. (2016) and Lee et al. (2016) solve SQuAD by ranking continuous text spans within passage. Yang et al. (2016) present a fine-grained gating mechanism to dynamically combine word-level and character-level representation and model the interaction between questions and passages. Wang et al. (2016) propose matching the context of passage with the question from multiple perspectives.

Different from the above models, we introduce self-matching attention in our model. It dynamically refines the passage representation by looking over the whole passage and aggregating evidence relevant to the current passage word and question, allowing our model make full use of passage information. Weightedly attending to word context has been proposed in several works. Ling et al.

(2015) propose considering window-based contextual words differently depending on the word and its relative position. Cheng et al. (2016) propose a novel LSTM network to encode words in a sentence which considers the relation between the current token being processed and its past tokens in the memory. Parikh et al. (2016) apply this method to encode words in a sentence according to word form and its distance. Since passage information relevant to question is more helpful to infer the answer in reading comprehension, we apply self-matching based on question-aware representation and gated attention-based recurrent networks. It helps our model mainly focus on question-relevant evidence in the passage and dynamically look over the whole passage to aggregate evidence.

Another key component of our model is the attention-based recurrent network, which has demonstrated success in a wide range of tasks. Bahdanau et al. (2014) first propose attention-based recurrent networks to infer word-level alignment when generating the target word. Hermann et al. (2015) introduce word-level attention into reading comprehension to model the interaction between questions and passages. Rocktäschel et al. (2015) and Wang & Jiang (2016a) propose determining entailment via word-by-word matching. The gated attention-based recurrent network is a variant of attention-based recurrent network with an additional gate to model the fact that passage parts are of different importance to the particular question for reading comprehension and question answering.

6 CONCLUSION

In this technical report, we present R-NET for reading comprehension and question answering. We introduce the gated attention-based recurrent networks and self-matching attention mechanism to obtain representation for the question and passage, and then use the pointer-networks to locate answer boundaries. Our model achieves state-of-the-art results on both SQuAD and MS-MARCO datasets, outperforming several strong competing systems. For future work, we will try to use syntax and knowledge base information into our system. Besides, we are also working on designing new network structures to handle questions that require complex inferences.

ACKNOWLEDGEMENT

We thank Pranav Samir Rajpurkar and Percy Liang for help in SQuAD submissions.

REFERENCES

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

Jonathan Berant, Vivek Srikumar, Pei-Chun Chen, Abby Vander Linden, Brittany Harding, Brad Huang, Peter Clark, and Christopher D. Manning. Modeling biological processes for reading comprehension. 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*, 2014.

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

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In *Association for Computational Linguistics (ACL)*, 2017.

Jianpeng Cheng, Li Dong, and Mirella Lapata. Long short-term memory-networks for machine reading. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, 2016.

Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 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 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pp. 1724–1734, 2014.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. Attention-over-attention neural networks for reading comprehension. *CoRR*, 2016.

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

Yichen Gong and Samuel R Bowman. Ruminating reader: Reasoning with gated multi-hop attention. *arXiv preprint arXiv:1704.07415*, 2017.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015*, pp. 1693–1701, 2015.

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

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

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

Kenton Lee, Tom Kwiatkowski, Ankur Parikh, and Dipanjan Das. Learning recurrent span representations for extractive question answering. *arXiv preprint arXiv:1611.01436*, 2016.

Wang Ling, Yulia Tsvetkov, Silvio Amir, Ramon Fermandez, Chris Dyer, Alan W. Black, Isabel Trancoso, and Chu-Cheng Lin. Not all contexts are created equal: Better word representations with variable attention. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, 2015.

Rui Liu, Junjie Hu, Wei Wei, Zi Yang, and Eric Nyberg. Structural embedding of syntactic trees for machine comprehension. *arXiv preprint arXiv:1703.00572*, 2017.

Christopher D Manning, Mihai Surdeanu, John Bauer, Jenny Rose Finkel, Steven Bethard, and David McClosky. The stanford corenlp natural language processing toolkit. In *ACL (System Demonstrations)*, pp. 55–60, 2014.

Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Interspeech*, 2010.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. MS MARCO: A human generated machine reading comprehension dataset. *CoRR*, abs/1611.09268, 2016.

Ankur P. Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. A decomposable attention model for natural language inference. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, 2016.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. Glove: Global vectors for word representation. 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*, pp. 1532–1543, 2014.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2016.

Matthew Richardson, Christopher J. C. Burges, and Erin Renshaw. Mctest: A challenge dataset for the open-domain machine comprehension of text. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 193–203, 2013.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomás Kociský, and Phil Blunsom. Reasoning about entailment with neural attention. *CoRR*, 2015.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. Bidirectional attention flow for machine comprehension. *arXiv preprint arXiv:1611.01603*, 2016.

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the Workshop on Cognitive Computation: Integrating neural and symbolic approaches 2016 co-located with the 30th Annual Conference on Neural Information Processing Systems (NIPS 2016), Barcelona, Spain, December 9, 2016.*, 2016.

Alessandro Sordoni, Phillip Bachman, and Yoshua Bengio. Iterative alternating neural attention for machine reading. *CoRR*, abs/1606.02245, 2016.

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*, 2014.

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

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. Pointer networks. In *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pp. 2692–2700, 2015.

Shuohang Wang and Jing Jiang. Learning natural language inference with LSTM. In *NAACL HLT 2016, The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, San Diego California, USA, June 12-17, 2016*, 2016a.

Shuohang Wang and Jing Jiang. Machine comprehension using match-lstm and answer pointer. *arXiv preprint arXiv:1608.07905*, 2016b.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. Gated self-matching networks for reading comprehension and question answering. In *Association for Computational Linguistics (ACL)*, 2017.

Zhiguo Wang, Haitao Mi, Wael Hamza, and Radu Florian. Multi-perspective context matching for machine comprehension. *arXiv preprint arXiv:1612.04211*, 2016.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. Fastqa: A simple and efficient neural architecture for question answering. *arXiv preprint arXiv:1703.04816*, 2017.

Caiming Xiong, Victor Zhong, and Richard Socher. Dynamic coattention networks for question answering. *arXiv preprint arXiv:1611.01604*, 2016.

Yi Yang, Wen-tau Yih, and Christopher Meek. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of EMNLP*, pp. 2013–2018. Citeseer, 2015.

Zhilin Yang, Bhuwan Dhingra, Ye Yuan, Junjie Hu, William W. Cohen, and Ruslan Salakhutdinov. Words or characters? fine-grained gating for reading comprehension. *CoRR*, abs/1611.01724, 2016.

Yang Yu, Wei Zhang, Kazi Hasan, Mo Yu, Bing Xiang, and Bowen Zhou. End-to-end reading comprehension with dynamic answer chunk ranking. *arXiv preprint arXiv:1610.09996*, 2016.

Matthew D. Zeiler. ADADELTA: an adaptive learning rate method. *CoRR*, abs/1212.5701, 2012.

Junbei Zhang, Xiaodan Zhu, Qian Chen, Lirong Dai, and Hui Jiang. Exploring question understanding and adaptation in neural-network-based question answering. *arXiv preprint arXiv:1703.04617*, 2017.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. Neural question generation from text: A preliminary study. *arXiv preprint arXiv:1704.01792*, 2017.