IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.DOI

R-Trans: RNN Transformer Network for Chinese Machine Reading Comprehension

SHANSHAN LIU[1], SHENG ZHANG[1], XIN ZHANG[1] and HUI WANG[1]

[1]Science and Technology on Information Systems Engineering Laboratory, College of Systems Engineering, National University of Defense Technology, Changsha 410073, China

Corresponding author: Xin Zhang (e-mail: ijunzhanggm@gmail.com).

ABSTRACT Machine reading comprehension (MRC) has gained increasingly wide attention over the past few years. A variety of benchmark datasets have been released, which triggers the development of quite a few MRC approaches based on deep learning techniques. However, most existing models are designed to address English MRC. When applying them directly to Chinese documents, the performance often degrades considerably because of some special characteristics of Chinese, the inevitable segmentation errors in particular. In this article, we present the RNN Transformer (R-Trans) network to tackle the Chinese MRC task. To mitigate the influence of incorrect word segmentation and mine sequential information of whole sentences, deep contextualized word representations and bidirectional Gated Recurrent Units networks are adopted in our model. Extensive experiments have been conducted on a very large scale Chinese MRC corpus, viz. the Les MMRC dataset. The results show that the proposed model outperforms the baseline and other prevalent MRC models notably, and established a new state-of-the-art record on the Les MMRC dataset.

INDEX TERMS contextualized word representation, deep learning, machine reading comprehension

I. INTRODUCTION

Machine reading comprehension (MRC), which empowers machine with the ability to answer questions based on the provided passage, has become an active area of research in natural language processing (NLP) in the past few years. Various deep neural models, e.g., Bi-DAF [1], R-Net [2], DCN [3] and Mnemonic Reader [4], have been proposed for English MRC and achieved impressing results.

Compared to English, there are few successful stories regarding Chinese MRC. We argue that this is mainly due to some special characteristics of Chinese. In particular, since there is no delimiter between Chinese words, word segmentation is often an indispensable step for further processing in Chinese NLP, which is itself a very challenging task. Usually, segmentation errors are unavoidable and would propagate to downstream tasks. For better understanding of this side-effect on Chinese MRC, Table 1 gives two such examples, in which the segmentation results are generated using the Jieba toolkit (https://pypi.org/project/jieba/). In Example 1, the character sequence "推特" is expected to be treated as a single word referring to *Twitter*, but Jieba segments it into "推" (push) and "特" (special), which have no relation with *Twitter* and would make it much more difficult for accurate understanding of the input question. In Example 2, the input question is indeed to ask people's evaluation of Brooks, and hence the character sequence "为什么" is supposed to be divided as "为(is)|什么(what)." However, Jieba identifies it as a single word meaning *why* in Chinese, which leads it nearly impossible to comprehend the true intent of the question.

The ambiguities caused by incorrect segmentations can be resolved using local and global context. In Example 1, the former and later word of "推特"(Twitter) are "脸书"(Facebook) and "微信"(Wechat), respectively, which constitute the context and both refer to social media platforms. If machine can mine this contextual information, it is easy to infer the meaning of "推特" even though there is an error made by segmentation toolkits. In Example 2, the meaning of whole sentence plays an important role to comprehend the true intent of the question. However, word2vec [5] or GloVe [6] which is commonly utilized in existing English MRC models is incapable of producing different vectors for

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901547, IEEE Access

TABLE 1. Two examples of incorrect segmentations

Example 1	
Input Question:	短片在脸书、推特、微信等社交媒体传播开后引发了什么
Segmentation Result:	短片|在|脸书|、|推|特|、|微信|等|社交|媒体|传播|开|后|引发|了|什么
	What happend after the short vedio spread through Facebook, Twitter and Wechat?
Example 2	
Input Question:	布鲁克斯被评为什么
Segmentation Result:	布鲁克斯|被评|为什么
	What do people think of Brooks?

[1] "|" is the delimiters between words and characters in red denote the incorrect segmentations.

one word according to context. Therefore, to mitigate the influence of incorrect segmentations to downstream tasks especially in Chinese MRC, more powerful representation learning approaches are in demand.

In addition, existing MRC models, e.g., QANet [7], suffer from two limitations. Firstly, Convolution Neural Network (CNN) is an essential component in QANet, but compared to Recurrent Neural Network (RNN) which is good at extracting global information, CNN can only mine local information. Secondly, these models often integrate many layers into their networks, which leads them to contain a large number of parameters, and makes the training and testing procedures cost very high computational resources. Thus, it is hard to operate them in some resource-constrained settings like mobile or embedded devices.

In this article, we propose RNN Transformer network (R-Trans) for Chinese MRC task which deals with the above problems in the following three aspects. At first, as examples in Table 1 illustrate, taking contextual information into account can relieve the effect of incorrect segmentation, but the distributed representations cannot change according to context. Hence deep contextualized word embeddings are added to the embedding layer to extract contextual information more effectively and contribute to the answer prediction. Secondly, to address the problem that CNN is not sufficient to extract global information, inspired by QANet, we retain the Transformer introduced by Vaswani et al. [8] and add bidirectional Gated Recurrent Units (GRU) networks to our R-Trans model. Thirdly, some network pruning is performed to cut down redundant parameters and accelerate training and inference.

Overall, the main contributions of this article are as follows:

- We incorporate deep contextualized word representation into the embedding layer and our case study demonstrates that it can mitigate the influence of errors made by segmentation toolkits.
- We add a bidirectional GRU network on top of the embedding layer to better extract global information. Furthermore, we prune the network and reduce the amount of parameters. Our R-Trans model, compared with QANet, not only improve the accuracy of the answer but also speed up training and inference.
- We conduct various experiments on a large scale Chinese MRC dataset, Les MMRC and experimental evaluations show that our R-Trans model achieves state-of-the-art results.

II. RELATED WORK

In the following, we give a brief introduction to the research closely related to our work, i.e., machine reading comprehension and word representation.

A. MACHINE READING COMPREHENSION

Machine reading comprehension (MRC), which involves a variety of complex techniques such as semantic understanding and knowledge reasoning, has become an important task of natural language processing and is far more challenging.

Benchmark datasets have stimulated recent progress in MRC. They can be roughly divided into four categories: multiple choice (McTest [9], ARC [10], RACE [11]), cloze-style (CNN/Daily News [12], CBT [13], CLOTH [14]), extractive (SQuAD [15], NewsQA [16], QuAC [17]) and abstractive (MS MARCO [18], NarrativeQA [19]).

End-to-end Neural Networks have featured predominantly in MRC and several deep learning models have been introduced to address this problem. Wang and Jiang [20] combine match-LSTM and Pointer Net to predict the boundary of the answer. Wang et al. [21] design a multi-perspective context matching model to identify the answer span by matching the passage with the question from multiple perspectives. The attention mechanism, which emphasizes important parts of the context related to the question, is widely used in MRC models. Seo et al. [1] apply the bidirectional attention flow mechanism to obtain the interactive information between questions and passages. Xiong et al. [3] propose a dynamic coattention network to iteratively estimate the answer span. Wang et.al [2] add gated attention-based recurrent networks to stress different importance of words in the passage to answer the particular question. To speed up training and inference, Yu et al. [7] substitute recurrent neural networks with convolution and self-attention. Different from models mentioned above, others are multi-pass reasoners which revisit the question and the passage more than one time. Shen et al. [22] introduce a termination state in the inference with the use of reinforcement learning, while Liu et al. [23] apply the stochastic dropout to the answer module which contributes to multi-step reasoning. Moreover, thinking that the extractive model is not sufficient, Tan et al. [24] develop an answer generation model to elaborate the final answers with the sub-spans predicted by the extraction model.

Methods using deep learning techniques have presented their advantage when addressing English MRC task. In con-

trast, there exists relatively little work on Chinese MRC due to the lack of high-quality Chinese dataset. The releasing of Baidu company's large-scale Chinese dataset, DuReader [4], sparks research interest into Chinese MRC. But due to features of DuReader, researchers pay much attention to addressing the problem of multi-passage MRC [25]–[27] , which selects the correct answer from a set of candidate passages for a question. In comparison, we focus on special characteristics of Chinese and propose more powerful model to tackle single-passage Chinese MRC task.

B. WORD REPRESENTATION

How to represent words to enable machine get their meanings is a basic task in natural language processing. Traditional methods use one-hot representation [28], a vector with all zeroes except in one position corresponding to the word, to encode individual words. However, such representations are sparse and high-dimensional when the vocabulary size is large. In addition, these methods cannot mine the relation between words.

To address the shortcomings, distributed word representation called word embeddings is proposed by Rumelhart et al. [29]. Various techniques to generate this representation have been introduced, among which the most popular ones are word2vec and GloVe. Mikolov et al. [5] propose two methods to represent words, CBOW and skip-gram, which are capable to represent words with continuous vectors in low dimension. GloVe put forward by Pennington et al. [6] combine co-occurrence matrix and local context window to efficiently obtain vector representations for words. Vectors produced by these methods have been widely applied in downstream NLP tasks, such as machine translation [30], name entity recognition [31], sentiment analysis [32] and dialogue systems [33].

Word representation also plays an important role in machine reading comprehension. As research of Dhingra et al. [34] shows, the minor choices made in word representation can lead to substantial differences in the final performance of the reader. The most common method to represent words in MRC models is to utilize both word-level and character-level embeddings, which maps each word to a vector space using pre-trained word embedding model [1], [2], [21], [35], [36]. However, this method seems not sufficient as it just simply concatenate word-level and character-level embeddings and vectors produced for one word cannot change according to context. To address these problems, Peters et al. [37] introduce deep contextualized word representations called ELMo which are pre-trained by language model first and fine-tuned according to downstream tasks. Seeing the limitation of unidirectional language models used in ELMo, Devlin et al. [38] propose BERT, which utilize bidirectional Transformer to encode both left and right context to representations.

Although word representations like ELMo and BERT show promising performances in various natural language processing tasks, they are scarcely applied to Chinese domain. Because of their contextualized features, those representations are thought to be helpful to mitigate the influence of incorrect word segmentation in Chinese documents. Considering that Chinese words have more abundant meanings than characters, we pre-train ELMo by ourselves based on Chinese words instead of using BERT released by Google based on Chinese characters in this article.

III. PROPOSED MODEL

Fig. 1 presents the framework of our R-Trans model. To be specific, our model has five layers, which are Input Embedding Layer, Embedding Encoder Layer, Context-Query Attention Layer, Model Encoder Layer and Output Layer and we will give an elaborate illustration in the following part.

A. INPUT EMBEDDING LAYER

Compared to common techniques which obtain representation of each word by concatenating its word embedding and character embedding, we add the results of ELMo encoder in addition.

In terms of the ELMo embeddings, we will give a more detailed description. We pre-train a bidirectional Language Model (biLM) with corpus of news collected from a variety of Internet pages for easy transfer learning. Given a sentence of N words, $(w_1, w_2, ..., w_N)$, we first compute a word representation x_k^{LM} independent of the context, which is the input of a multiple-layer bidirectional Long Short Term Memory (biLSTM) with residual connections. At each position k, a forward LSTM layer generates a context-dependent representation $\overrightarrow{h}_{k,i}^{\mathrm{LM}}$ where $i = 1, ..., L$, while a backward one outputs $\overleftarrow{h}_{k,i}^{\mathrm{LM}}$ in a similar way. Thus, each word w_k has $2L+1$ representations through a L-layer biLM.

$$\begin{aligned} R_k &= \{x_k^{\mathrm{LM}}, \overrightarrow{h}_{k,i}^{\mathrm{LM}}, \overleftarrow{h}_{k,i}^{\mathrm{LM}} | i = 1, ..., L\} \\ &= \{h_{k,i}^{\mathrm{LM}} | i = 1, ..., L\} \end{aligned} \tag{1}$$

ELMo projects outputs of all layers in the biLM into a single vector by a linear combination

$$\begin{aligned} \mathrm{s} &= \mathrm{softmax(w)} \\ \mathrm{ELMo}_k &= \gamma \sum_{i=0}^{L} s_i h_{k,i}^{\mathrm{LM}} \end{aligned} \tag{2}$$

where γ and w are parameters which are supposed to be tuned according to the specific task during training. Thus, we get the ELMo embedding of given word denoted as x_e.

Then, we pre-train the word into a 300-dimension vector with GloVe [6] and then store the results in the embedding. Namely, we look up this table in Input Embedding Layer to get the word embedding for each word. As for the character embeddings, Convolutional Neural Networks (CNN) are applied. Each character in the word is embedded into a 64-dimension vector, which is fed to the CNN as 1D inputs. After max-pooling the entire width, the outputs are the results of character embeddings, a fixed-size vector for the word.

At last, for the given word x, its word embedding x_w, character embedding x_c and ELMo embedding x_e are con-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901547, IEEE Access

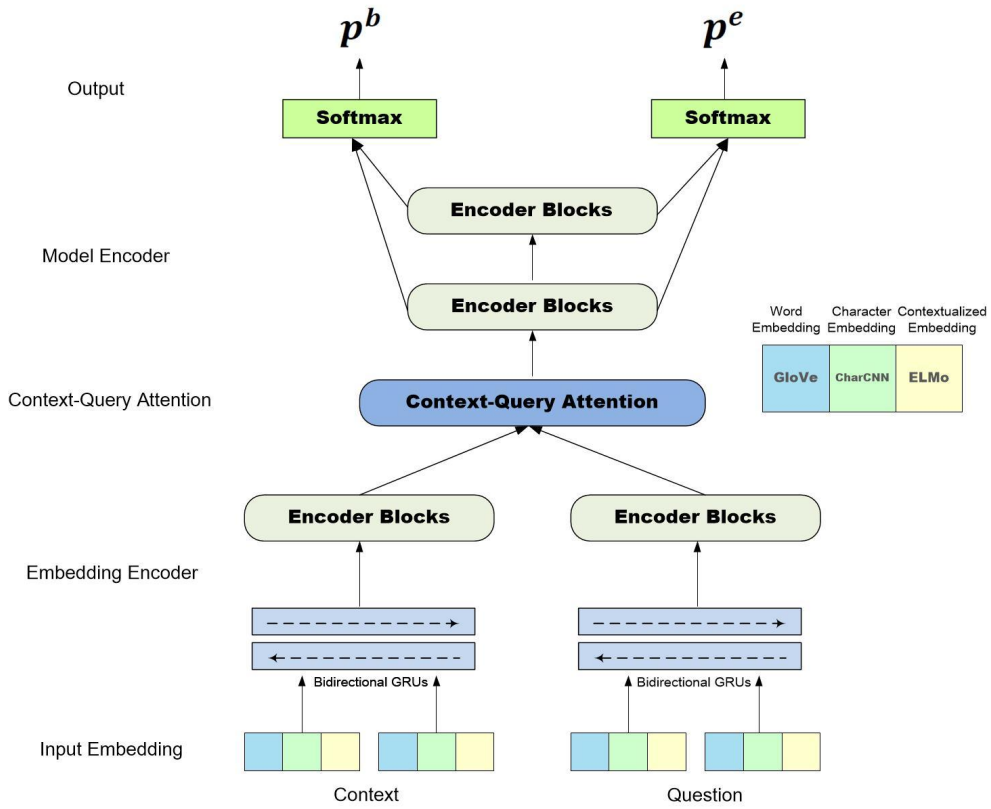


FIGURE 1. The framework of R-Trans model.

catenated together and through a two-layer highway network, the representation of word can be denoted as $[x_w; x_c; x_e]$.

B. EMBEDDING ENCODER LAYER

QANet removes all RNN in previous models with convolution and self-attention for sake of speed-up, but convolution operations pay much attention to local information and cannot deal with sequential text as well as RNN. Hence, we add a bidirectional Gated Recurrent Units network (GRU) [30], a variant of RNN, on top of the embeddings to encode temporal interactions between words and concatenate the outputs of two GRUs at each time step. The outputs of bidirectional GRUs can be denoted as follows:

$$\begin{aligned} \overrightarrow{h}_t &= \text{GRU}[x_w; x_c; x_e] \\ \overleftarrow{h}_t &= \text{GRU}[x_w; x_c; x_e] \\ H_t &= [\overrightarrow{h}_t; \overleftarrow{h}_t] \end{aligned} \tag{3}$$

where $\overrightarrow{h}_t$ is the output of forward GRU and $\overleftarrow{h}_t$ is the output of backward one.

Then the outputs of bidirectional GRUs are fed to the residual block, a stack of the Encoder Block, to alleviate the impact of gradient explosion and vanishing. Based on the architecture of Transformer introduced by Vaswani et al. [8], the Encoder Block adds convolution layer before self-attention layer and feed-forward layer. Rather than traditional convolutions, the depthwise separable ones are utilized for their efficient memory and better generalization. For self-attention layer, we change the number of heads to 1, which simplify the multi-head attention mechanism to a scaled dot-product attention. The number of the Encoder Block in residual block is 1.

C. CONTEXT-QUERY ATTENTION LAYER

This layer, which obtains interactive information between context and question by computing both context-to-query attention and query-to-context attention, is common in most existing machine reading comprehension models.

The encoded context and question are denoted as C and Q respectively. We apply the trilinear function to compute the similarity between context and question. The similarity matrix $S \in \mathbb{R}^{I \times J}$ is computed by

$$S_{ij} = \alpha(C_i, Q_j) \tag{4}$$

where α is the trilinear function, C_i is i-th context word and Q_j is j-th question word.

After normalizing each row of S with softmax function, the attended question vectors for the whole context can be

calculated as

$$A = \text{softmax}(S)Q^T \tag{5}$$

To signify which context words are critical to answer the question because of high similarity with question words, the query-to-context attention is calculated as:

$$B = \text{softmax}(S^T)C^T \tag{6}$$

here we use softmax function to normalize each column of S.

D. MODEL ENCODER LAYER

Compared to original QANet model, we reduce the number of model encoder block to 2. Besides, for each model encoder block, the number of the Encoder Block introduced above decreases from 7 to 3. The input of this layer is denoted as $[c, a, c \odot a, c \odot b]$, where $\odot$ is the element-wise multiplication, a and b are a row of matrix A and B respectively. Different from the Encoder Block applied in the Embedding Encoder Layer, the number of convolution layer is 2 and the kernel size is 5.

E. OUTPUT LAYER

The goal of this layer is to predict the probability of each position in the given passage to be the begin or end of the answer. To achieve this goal, we combine the outputs of two model encoder blocks in previous layer, denoted as M_0 and M_1 respectively, and the probabilities are computed as:

$$\begin{aligned} p^b &= \text{softmax}(W_1[M_0; M_1]) \\ p^e &= \text{softmax}(W_2[M_0; M_1]) \end{aligned} \tag{7}$$

where W_1 and W_2 are trainable parameters. Like the previous reading comprehension models, we minimize the sum of the negative log probabilities of the true begin and end position by the predicted distributions:

$$L(\theta) = -\frac{1}{N}\sum_{i}^{N}[\log(p^b_{y_i^1}) + \log(p^e_{y_i^2})] \tag{8}$$

where y_i^1 and y_i^2 are the ground-truth begin and end position of example i.

IV. EXPERIMENTS

In this section, we introduce our experiments in detail. To begin with, we give a description of the dataset and metrics. Data pre-processing and experimental settings are illustrated next. In the end, we talk about the results.

A. DATASET

The dataset used in this article is Les MMRC, a large scale Chinese machine reading comprehension dataset, which is released by the 28th Research Institute of China Electronics Technology Group Corporation. Les MMRC consists of more than 50,000 news articles collected from a variety of Internet pages. For each passage, about five questions are asked thus the dataset contains 250,000 questions or so in total, including 200,000 for the training set and 50,000 for the test set. Different from the test set, questions in the training set have ground-truth answers responded by human. In this article, to evaluate different training strategies, we select 10% of data from the training set randomly as the development set.

Passage	
（早报讯）菲律宾一架小型飞机周六起飞后不久，不知何故失控坠落并撞入民宅，导致机上五人和地面两人丧命…… (Morning News) A plane lost control and fell into the house shortly after taking off causing that five people on board and two people on the ground died……	
Question	**Answer**
几人丧命	7
How many people died	seven

FIGURE 2. An example of the question which cannot be answer by a span in the original passage.

Types of questions in Les MMRC are various, asking about opinion, fact, definition, number and so on. Most of that can be answered by a span in original passage, just like SQuAD. However, it is more challenging that some answers to the questions need inference and conclusion, which cannot be found in the passage directly. As the example presented in Fig. 2, to answer the question *how many people died*, the machine needs to inference from the original passage segmentation that *five people on board and two people on the ground died* to give the right answer *seven*.

B. METRICS

In this article, two metrics, Rouge-L and BLEU-4, are used as performance evaluation indicators for the model.

Rouge (Recall-Oriented Understudy for Gisting Evaluation) is commonly used in various natural language processing tasks like machine translation and summarization. L in Rouge-L is the initial of LCS, short for the longest common subsequence, which means Rouge-L applies the longest common subsequence to measure the similarity between text. Rouge-L can be calculated as follows:

$$R_{\text{lcs}} = \frac{\text{LCS}(X, Y)}{m} \tag{9}$$

$$P_{\text{lcs}} = \frac{\text{LCS}(X, Y)}{n} \tag{10}$$

$$F_{\text{lcs}} = \frac{(1+\beta)^2 R_{\text{lcs}} P_{\text{lcs}}}{R_{\text{lcs}} + \beta^2 P_{\text{lcs}}} \tag{11}$$

where $\text{LCS}(X, Y)$ is the length of the longest common subsequence of X and Y, m and n are the length of reference

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901547, IEEE Access

text and candidate text respectively, R_{lcs} , P_{lcs} represent recall and precision while F_{lcs} is the score of Rouge-L.

BLEU (Bilingual Evaluation Understudy) is usually used to evaluate the quality of machine translation by measuring the similarity between the sentences translated by machine and human. The more similar, the higher BLEU score. The calculations are as follows:

At first, we calculate the probability of the N-gram in candidate sentence appears in reference sentence:

$$P_n = \frac{\sum_{C \in \{\text{Candidate}\}} \sum_{\text{N-gram} \in C} \text{Count}_{\text{clip}}(\text{N-gram})}{\sum_{C' \in \{\text{Candidate}\}} \sum_{\text{N-gram}' \in C'} \text{Count}(\text{N-gram})} \quad (12)$$

If the length of candidate sentence is too short, the accuracy of BLEU score will decrease. To alleviate that, the penalty factor BP is introduced, which can be calculated as below:

$$\text{BP} = \begin{cases} 1, c > r \\ e^{1-\frac{r}{c}}, c \leq r \end{cases} \quad (13)$$

Finally, the BLEU is computed as:

$$\text{BLEU} = \text{BP} \cdot \exp(\sum_{n=1}^{N} w_n \log p_n) \quad (14)$$

when calculating BLEU-4, N equals 4 and w_n is $\frac{1}{4}$.

C. BASELINES

To prove the superiority of our model, we compare it with other baselines, which are listed as below:

QANet: QANet [7], proposed by Adams Wei Yu et al., replaces recurrent neural networks, which are primarily used in current machine reading comprehension models because of their sequential features, with convolution in the first place. Compared to prevalent recurrent models, it not only achieves equivalent accuracy on SQuAD, but also is faster in training and inference. With data augmentation technique, QANet reaches a competitive performance on SQuAD.

R-Net: This model [2] proposed by the Microsoft Research Asia, pays more attention to the interaction between questions and the passage by adding a gated attention-based recurrent networks. To aggregate evidence from the whole passage to infer the answer, a self-matching mechanism is also introduced. R-Net achieves better results on the SQuAD and MS MARCO than former models.

D. DATA PRE-PROCESSING

Data pre-processing plays a significant role in the machine reading comprehension task. As the data in original dataset may be noisy, invalid or duplicate, and the structure of that is defined by the database initially, it is difficult to train the models directly using the data without pre-processing. As a result, we have to do some data pre-processing before training.

1) Passage Pruning

Firstly, we do some common pre-processing, including replacing Chinese punctuation and number with English ones, deleting special characters which appear at the beginning and end of the answers and removing invalid data whose passage, question or answer is vacant and reduplicative one whose questions and answers for same passage are repeating.

After doing that, there is still an apparent problem in original dataset that some passages are too long. According to the data analysis of Les MMRC, about half of passages have more than 500 words, which will lead to the increase of parameters and in turn make it hard to train the model. Therefore, how to prune long passages to the limited length on the condition that information loss is as small as possible is an important task in MRC pre-processing.

In this article, the strategies applied to prune long passages are as follows:

At first, we define the upper limited length of passage as L taking features of dataset into consideration. If the length of passage is shorter than L, there is no need to prune, otherwise truncation is divided into two cases according to whether passages are segmented or not:

- If the passage is segmented, we apply the longest common substring to measure the similarity between questions and paragraphs. The more similar the question and paragraph are, the higher score the paragraph gets. The first k paragraphs with the highest score are selected in descending order to be concatenated as the new passage if the length of that does not exceed the limit. As is known that the first sentence usually contains more information than others in the paragraph, we add the first sentence of each paragraph to the processed passage provided that the length is within the upper limit.
- If the passage is not segmented, like the strategy introduced above, we also use the longest common substring to calculate the similarity between the question and each sentence at first, and then concatenate the most similar t sentences as the new passage without changing their order.

2) Answer Labelling

The model in this article is extractive, which assumes that the answer corresponds to a span in the passage and the task is to predict this span. Although there are passages, questions and their corresponding answers in the training set, the locations of answers in the passage are not labelled. As the task of MRC is supervised, we need to label the answer in pre-processing.

We match the answer with the original passage, and there are three cases according to the number of matches:

- If the number of matches is one, we directly label that location.
- When the number of matches is more than one, it is supposed that the answer is inclined to appear in the part of passage which is more similar to the question. Hence,

we label the location which is near to the begin or end of the longest common substring of passage and question.
- Irregularly labelled answers which have redundant spaces or incorrect punctuation result in missing matches with the passage. To label those answers, we calculate the longest common subsequence of the question and passage, whose length is more than 70 percent of the answer's length will be labelled as the location of that answer.

According to answer labelling, we know that the part of the passage, which is not labelled, has nothing to do with the prediction generally, so the passage can be further pruned. If the end label of the answer is within the upper limit L, the part after the end label can be pruned. Otherwise, we truncate the text from title to the begin label of the answer randomly and then remove the text after the end label which is out of range.

E. EXPERIMENTAL SETTINGS

When training the models, the parameters are set as presented in Table 2. We save both the best model with the highest Rouge-L score and the last one before terminating the training process to make it easy for training continuously.

TABLE 2. Parameters settings utilized in R-Trans model

Parameters Name	Illustration	Value
glove_dim	the dimension of word embedding	300
char_dim	the dimension of character embedding	64
para_limit	the length of passage(train)	400
ques_limit	the length of question(train)	50
ans_limit	the length of answer(train)	75
test_para_limit	the length of passage(test)	1,000
test_ques_limit	the length of question(test)	100
batch_size	the size of batch	32
num_steps	training steps	140,000
checkpoint	checkpoint	2,000

[1] The length is in word granularity.

F. RESULTS

Table 3 illustrates the evaluation results of several models on Les MMRC. The first column is the index and the second one is the models in this experiment. The third and fourth columns present Rouge-L score and BLEU-4 score on the test set separately.

Row 1 to 3 show the results of models whose word representation just use the concatenation of character and word embeddings in the traditional way. Although the original QANet model outperforms the R-Net in both Rouge-L and BLEU-4 score, our R-Trans model achieves the best performance among these three models, improving Rouge-L and BLEU-4 score by (1.28%, 0.94%) compared to the original QANet.

TABLE 3. The evaluation results of diffierent models on Les MMRC

ID	Model	Rouge-L(%)	BLEU-4(%)
1	QANet	88.16	80.84
2	R-Net	87.96	80.82
3	R-Trans	**89.44**	**81.78**
4	QANet+POS	90.09	83.79
5	QANet+ELMo	90.68	84.71
6	R-Net+ELMo	88.05	81.79
7	R-Trans+ELMo	**90.77**	**85.35**

Models in row 4 to 7 apply other word representation strategies like ELMo and part-of-speech tags (POS tags). The inclusion of POS tags is to show that richer word representations contribute to higher accuracy of answer prediction. Both POS tags and ELMo can improve the performance of models while ELMo is more powerful as Rouge-L score and BLEU-4 score of QANet utilized ELMo are higher than QANet with POS tags. Compared to the results in row 1 to 3, row 5 to 7 illustrate that ELMo indeed makes a contribution to different downstream models, however, its effort differs from models. As can be seen in the last row, our R-Trans model with ELMo reaches the highest Rouge-L and BLEU-4 score, which indicates that more sematic information can be mined owing to the contextualized word representation and in turn enhance the accuracy of answer prediction.

G. DISCUSSION

1) Comparative Analysis with QANet

To illustrate the superiority of our model to its baseline QANet, we conduct comparative analysis in this part. We use the same experimental environment and parameters settings to compare the amounts of parameters, train time and inference time of our model and QANet. The result is shown in Table 4 and presents that our model is 1.84 and 2.50 faster than QANet in training and inference speed with cutting down nearly 400,000 parameters.

TABLE 4. Speed and parameters comparision results

	Parameters	Training	Inference
QANet	1,719,233	4h37min	10min
R-Trans	1,323,329	2h30min	4min
Reduction/Speedup	**1.38x**	**1.84x**	**2.50x**

[1] Train time means how much time spends on getting 88.0 Rouge-L on Dev set.

We plot the Rouge-L/BLEU-4 score on Les MMRC test set in Fig. 3 for R-Trans and QANet. The horizontal axis is number of iteration during training while the vertical axis is Rouge-L/BLEU-4 score. We can see that both Rouge-L and BLEU-4 curves of our model not only converge faster but also reach higher values than that of QANet, showing better performance of our R-Trans in predicting answers.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901547, IEEE Access

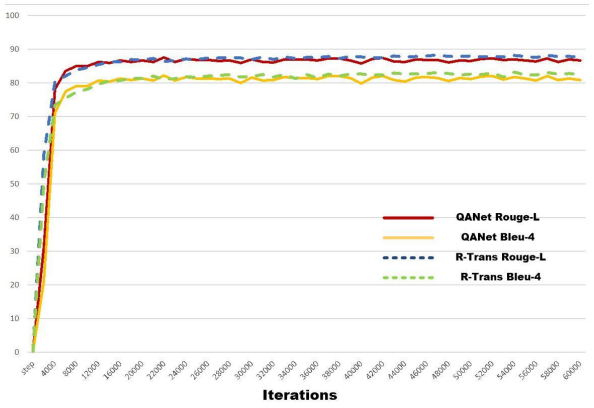


FIGURE 3. The Rouge-L and BLEU-4 curves of QANet and R-Trans while training.

2) Ablation Analysis

We conduct ablation studies mainly on components of Input Embedding Layer of our proposed model in order to evaluate the performance of different representation methods.

As shown in Table 5, the first row presents the result of our model, and its ablations are listed below. We can see from Row 2 to 4 that word-level, character-level embeddings and ELMo all contribute to the model's performance while ELMo is more crucial. Ablating ELMo results a performance drop of 1.49% and 3.63% on Rouge-L and BLEU-4, respectively, which is far greater than the degradation caused by removing word level or character level embeddings.

However, word level and character level embeddings are indeed indispensable. From Row 5 to Row 7, we remove two components each time to investigate the effort of single representation. To be concrete, removing both ELMo and word level embeddings means we only use character level embeddings in Input Embedding Layer. The ablation of word level and character level embeddings accounts for 2.57% and 4.18% on Rouge-L and BLEU-4, which indicates that just applying ELMo to represent words is not sufficient.

We interpret these phenomena as follows: the word level embeddings can represent the semantics of each word while the character level embeddings are good at handling out-of-vocabulary (OOV) words. Moreover, ELMo, which is deep contextualized, can encode contextual information effectively and help to relieve the ambiguity caused by incorrect word segmentation in Chinese text. As they cannot replace each other, combining three of them together makes the most contribution to upgrading the performance.

3) Case Study

To demonstrate how contextualized word representation influences the prediction of final answers, we conduct a case study in Table 6 with three examples selected from the outputs of our R-Trans model.

In Example 1, Jieba treats "名额"(quota) as a single word, leading to a wrong answer *Hundreds of places* given by model without ELMo. However, model with ELMo is able to infer from surrounding context that the question asks about *How many additional people* and gives the right answer *Hundreds of people* in spite of segmentation errors. In Example 2, we can see that model with ELMo gets the intent of question and answers correctly while model without ELMo is misled by erroneous segmentation and answers with a reason. In Example 3, contextualized word representations show their advantage in controlling the boundary of the answer. Although "千比特"(kilobit) is segmented as "千(thousand)|比特(bite)," model with ELMo adds the missing part "比特" compared to the answer given by model without ELMo, which makes the syntax of sentence complete.

To summarize, effectively encoding contextual information indeed contributes to mitigating the influence of incorrect word segmentations in Chinese documents and upgrading the downstream answer prediction.

TABLE 5. Ablation analysis results

	Components	Rouge-L(%)	BLEU-4(%)
1	R-Trans+char+word+ELMo	90.77	85.35
2	-ELMo	**-1.49**	**-3.63**
3	-word	-0.5	-1.07
4	-char	-0.4	-1.25
5	-ELMo-word	-1.79	-3.84
6	-ELMo-char	-2.39	-3.33
7	-word-char	**-2.57**	**-4.18**

V. CONCLUSION

In this article, we introduce R-Trans model with deep contextualized word representation to address the problems in Chinese MRC task. Our model not only spends less time in training and inference but also mitigates the influence made by erroneous segmentation to downstream answer prediction. As experimental results demonstrate, our R-Trans model surpasses baseline models with state-of-the-art performance on Les MMRC.

After analyzing the answer responded by our model, we find that questions which need inference and conclusion cannot be correctly answered in general. That may be the weakness of extractive MRC models. Our future work would like to introduce external knowledge to our model by constructing knowledge graph to achieve better results.

REFERENCES

[1] M. Seo, A. Kembhavi, A. Farhadi, and H. Hajishirzi, "Bidirectional attention flow for machine comprehension," arXiv preprint arXiv:1611.01603, 2016.

[2] W. Wang, N. Yang, F. Wei, B. Chang, and M. Zhou, "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: Long Papers), vol. 1, 2017, pp. 189–198.

[3] C. Xiong, V. Zhong, and R. Socher, "Dynamic coattention networks for question answering," arXiv preprint arXiv:1611.01604, 2016.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2901547, IEEE Access

TABLE 6. Case study

Example 1	
Question:	庞塞号上的简易铺位还可以容纳多少名额外人员
Segmentation:	庞塞号|上|的|简易|铺位|还|可以|容纳|多少|名额|外|人员
	How many additional people can be accommodated in the simple bunk on the Ponce?
Answer with ELMo:	数百名
	Hundreds of people.
Answer without ELMo:	数百名额
	Hundreds of quotas.
Example 2	
Question:	布鲁克斯被评为什么
Segmentation:	布鲁克斯|被评|为什么
	What do people think of Brooks?
Answer with ELMo:	精通掌握中国和朝鲜半岛军事动向
	Brooks is regarded as Mr. Know-all who is proficient in the military trends of China and the Korean Peninsula.
Answer without ELMo:	布鲁克斯曾在德国、韩国和科索沃工作，还参与了阿富汗和伊拉克战争
	Brooks worked in Germany, South Korea and Kosovo and also participated in the wars in Afghanistan and Iraq.
Example 3	
Question:	改进型保密航路通信套件于2002-2003年提供了每秒64千比特，后来提高到多少
Segmentation:	改进型|保密|航路|通信|套件|于|2002|-|2003|年|提供|了|每秒|64|千|比特|，|后来|提高|到|多少
	The improved Secure Route Communication Kit reached 64 kilobits per second in 2002-2003,how much was it later increased to?
Answer with ELMo:	每秒256千比特
	256 kilobits per second
Answer without ELMo:	每秒256千
	256 thousand per second

[4] W. He, K. Liu, J. Liu, Y. Lyu, S. Zhao, X. Xiao, Y. Liu, Y. Wang, H. Wu, Q. She et al., "Dureader: a chinese machine reading comprehension dataset from real-world applications," arXiv preprint arXiv:1711.05073, 2017.

[5] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, "Distributed representations of words and phrases and their compositionality," in Advances in neural information processing systems, 2013, pp. 3111–3119.

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

[7] A. W. Yu, D. Dohan, M.-T. Luong, R. Zhao, K. Chen, M. Norouzi, and Q. V. Le, "Qanet: Combining local convolution with global self-attention for reading comprehension," arXiv preprint arXiv:1804.09541, 2018.

[8] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," 2017.

[9] M. Richardson, C. J. Burges, and E. 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, 2013, pp. 193–203.

[10] P. Clark, I. Cowhey, O. Etzioni, T. Khot, A. Sabharwal, C. Schoenick, and O. Tafjord, "Think you have solved question answering? try arc, the ai2 reasoning challenge," 2018.

[11] G. Lai, Q. Xie, H. Liu, Y. Yang, and E. Hovy, "Race: Large-scale reading comprehension dataset from examinations," 2017.

[12] K. M. Hermann, T. Kocisky, E. Grefenstette, L. Espeholt, W. Kay, M. Suleyman, and P. Blunsom, "Teaching machines to read and comprehend," in Advances in Neural Information Processing Systems, 2015, pp. 1693–1701.

[13] F. Hill, A. Bordes, S. Chopra, and J. Weston, "The goldilocks principle: Reading children's books with explicit memory representations," arXiv preprint arXiv:1511.02301, 2015.

[14] Q. Xie, G. Lai, Z. Dai, and E. Hovy, "Large-scale cloze test dataset created by teachers," 2017.

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

[16] A. Trischler, T. Wang, X. Yuan, J. Harris, A. Sordoni, P. Bachman, and K. Suleman, "Newsqa: A machine comprehension dataset," 2016.

[17] E. Choi, H. He, M. Iyyer, M. Yatskar, W. T. Yih, Y. Choi, P. Liang, and L. Zettlemoyer, "Quac : Question answering in context," 2018.

[18] T. Nguyen, M. Rosenberg, S. Xia, J. Gao, and D. Li, "Ms marco: A human generated machine reading comprehension dataset," 2016.

[19] T. Kočiský, J. Schwarz, P. Blunsom, C. Dyer, K. M. Hermann, G. Melis, and E. Grefenstette, "The narrativeqa reading comprehension challenge," 2017.

[20] S. Wang and J. Jiang, "Machine comprehension using match-lstm and answer pointer," arXiv preprint arXiv:1608.07905, 2016.

[21] Z. Wang, H. Mi, W. Hamza, and R. Florian, "Multi-perspective context matching for machine comprehension," arXiv preprint arXiv:1612.04211, 2016.

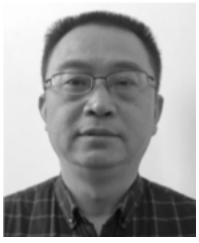
[22] Y. Shen, P.-S. Huang, J. Gao, and W. Chen, "Reasonet: Learning to stop

reading in machine comprehension," in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM, 2017, pp. 1047–1055.
[23] X. Liu, Y. Shen, K. Duh, and J. Gao, "Stochastic answer networks for machine reading comprehension," arXiv preprint arXiv:1712.03556, 2017.
[24] C. Tan, F. Wei, N. Yang, B. Du, W. Lv, and M. Zhou, "S-net: From answer extraction to answer generation for machine reading comprehension," arXiv preprint arXiv:1706.04815, 2017.
[25] Z. Li, J. Xu, Y. Lan, J. Guo, Y. Feng, and X. Cheng, "Hierarchical answer selection framework for multi-passage machine reading comprehension," in China Conference on Information Retrieval. Springer, 2018, pp. 93–104.
[26] M. Yan, J. Xia, C. Wu, B. Bi, Z. Zhao, J. Zhang, L. Si, R. Wang, W. Wang, and H. Chen, "A deep cascade model for multi-document reading comprehension," arXiv preprint arXiv:1811.11374, 2018.
[27] J. Liu, W. Wei, M. Sun, H. Chen, Y. Du, and D. Lin, "A multi-answer multi-task framework for real-world machine reading comprehension," in Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, 2018, pp. 2109–2118.
[28] R. Baeza-Yates, B. d. A. N. Ribeiro et al., Modern information retrieval. New York: ACM Press; Harlow, England: Addison-Wesley,, 2011.
[29] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," nature, vol. 323, no. 6088, p. 533, 1986.
[30] K. Cho, B. Van Merriënboer, C. Gulcehre, D. Bahdanau, F. Bougares, H. Schwenk, and Y. Bengio, "Learning phrase representations using rnn encoder-decoder for statistical machine translation," arXiv preprint arXiv:1406.1078, 2014.
[31] J. P. Chiu and E. Nichols, "Named entity recognition with bidirectional lstm-cnns," arXiv preprint arXiv:1511.08308, 2015.
[32] P. Nakov, A. Ritter, S. Rosenthal, F. Sebastiani, and V. Stoyanov, "Semeval-2016 task 4: Sentiment analysis in twitter," in Proceedings of the 10th international workshop on semantic evaluation (semeval-2016), 2016, pp. 1–18.
[33] I. V. Serban, A. Sordoni, Y. Bengio, A. C. Courville, and J. Pineau, "Building end-to-end dialogue systems using generative hierarchical neural network models." in AAAI, vol. 16, 2016, pp. 3776–3784.
[34] B. Dhingra, H. Liu, R. Salakhutdinov, and W. W. Cohen, "A comparative study of word embeddings for reading comprehension," arXiv preprint arXiv:1703.00993, 2017.
[35] D. Weissenborn, G. Wiese, and L. Seiffe, "Fastqa: A simple and efficient neural architecture for question answering," CoRR, abs/1703.04816, 2017.
[36] M. Hu, Y. Peng, Z. Huang, X. Qiu, F. Wei, and M. Zhou, "Reinforced mnemonic reader for machine reading comprehension," arXiv preprint arXiv:1705.02798, 2017.
[37] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," arXiv preprint arXiv:1802.05365, 2018.
[38] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint arXiv:1810.04805, 2018.

SHANSHAN LIU was born in Gansu, China in 1994. She received the Bachelor of Administration degree from School of Information Management, Nanjing University in 2017. She is now a post-graduate student in National University of Defense Technology and is pursuing the Master's degree in Management Science and Engineering. Her research interests in natural language processing, data mining and social computing.

SHENG ZHANG received his B.S. degree in systems engineering from National University of Defense Technology (NUDT), Changsha, China, in 2015, and the M.S. degree in management science and engineering from NUDT in 2017. He is currently pursuing the Ph.D. degree in the College of Systems Engineering, NUDT, Changsha. His research interests include natural language processing, deep learning, and data mining.

XIN ZHANG received his B.S and Ph.D. degrees in system engineering from National University of Defense Technology (China), in 2000 and 2006, respectively. He is currently a professor with the State Key Lab of Information System Engineering, College of Systems Engineering, National University of Defense Technology. His research interests include cross-modal data mining, information extraction, and event analysis.



HUI WANG received his B.S, M.S. and Ph.D. degrees in system engineering from National University of Defense Technology (China), in 1990, 1998 and 2005, respectively. He is currently a professor with the State Key Lab of Information System Engineering, College of Systems Engineering, National University of Defense Technology. His research interests include natural language processing, deep learning, data mining, and social analysis.