Multi-Passage Machine Reading Comprehension with Cross-Passage Answer Verification

Yizhong Wang[1][*], Kai Liu[2], Jing Liu[2], Wei He[2],
Yajuan Lyu[2], Hua Wu[2], Sujian Li[1] and Haifeng Wang[2]

[1]Key Laboratory of Computational Linguistics, Peking University, MOE, China
[2]Baidu Inc., Beijing, China
{yizhong, lisujian}@pku.edu.cn, {liukai20, liujing46,
hewei06, lvyajuan, wu_hua, wanghaifeng}@baidu.com

Abstract

Machine reading comprehension (MRC) on real web data usually requires the machine to answer a question by analyzing multiple passages retrieved by search engine. Compared with MRC on a single passage, multi-passage MRC is more challenging, since we are likely to get multiple confusing answer candidates from different passages. To address this problem, we propose an end-to-end neural model that enables those answer candidates from different passages to verify each other based on their content representations. Specifically, we jointly train three modules that can predict the final answer based on three factors: the answer boundary, the answer content and the cross-passage answer verification. The experimental results show that our method outperforms the baseline by a large margin and achieves the state-of-the-art performance on the English MS-MARCO dataset and the Chinese DuReader dataset, both of which are designed for MRC in real-world settings.

1 Introduction

Machine reading comprehension (MRC), empowering computers with the ability to acquire knowledge and answer questions from textual data, is believed to be a crucial step in building a general intelligent agent (Chen et al., 2016). Recent years have seen rapid growth in the MRC community. With the release of various datasets, the MRC task has evolved from the early cloze-style test (Hermann et al., 2015; Hill et al., 2015) to answer extraction from a single passage (Rajpurkar et al.,

2016) and to the latest more complex question answering on web data (Nguyen et al., 2016; Dunn et al., 2017; He et al., 2017).

Great efforts have also been made to develop models for these MRC tasks , especially for the answer extraction on single passage (Wang and Jiang, 2016; Seo et al., 2016; Pan et al., 2017). A significant milestone is that several MRC models have exceeded the performance of human annotators on the SQuAD dataset[1] (Rajpurkar et al., 2016). However, this success on single Wikipedia passage is still not adequate, considering the ultimate goal of reading the whole web. Therefore, several latest datasets (Nguyen et al., 2016; He et al., 2017; Dunn et al., 2017) attempt to design the MRC tasks in more realistic settings by involving search engines. For each question, they use the search engine to retrieve multiple passages and the MRC models are required to read these passages in order to give the final answer.

One of the intrinsic challenges for such multi-passage MRC is that since all the passages are question-related but usually independently written, it's probable that multiple confusing answer candidates (correct or incorrect) exist. Table 1 shows an example from MS-MARCO. We can see that all the answer candidates have semantic matching with the question while they are literally different and some of them are even incorrect. As is shown by Jia and Liang (2017), these confusing answer candidates could be quite difficult for MRC models to distinguish. Therefore, special consideration is required for such multi-passage MRC problem.

In this paper, we propose to leverage the answer candidates from different passages to verify the final correct answer and rule out the noisy incorrect answers. Our hypothesis is that the cor-

---

[1]https://rajpurkar.github.io/SQuAD-explorer/

Question: What is the difference between a mixed and pure culture?
Passages:
[1] **A culture is a society's total way of living and a society is a group that live in a defined territory and participate in common culture.** While the answer given is in essence true, societies originally form for the express purpose to enhance ...
[2] ... There has been resurgence in the economic system known as capitalism during the past two decades. 4. **The mixed economy is a balance between socialism and capitalism.** As a result, some institutions are owned and maintained by ...
[3] **A pure culture is one in which only one kind of microbial species is found whereas in mixed culture two or more microbial species formed colonies.** Culture on the other hand, is the lifestyle that the people in the country ...
[4] Best Answer: **A pure culture comprises a single species or strains. A mixed culture is taken from a source and may contain multiple strains or species.** A contaminated culture contains organisms that derived from some place ...
[5] ... It will be at that time when we can truly obtain a pure culture. **A pure culture is a culture consisting of only one strain.** You can obtain a pure culture by picking out a small portion of the mixed culture ...
[6] **A pure culture is one in which only one kind of microbial species is found whereas in mixed culture two or more microbial species formed colonies.** A pure culture is a culture consisting of only one strain. ...
... ...
Reference Answer: A pure culture is one in which only one kind of microbial species is found whereas in mixed culture two or more microbial species formed colonies.

Table 1: An example from MS-MARCO. The text in bold is the predicted answer candidate from each passage according to the boundary model. The candidate from [1] is chosen as the final answer by this model, while the correct answer is from [6] and can be verified by the answers from [3], [4], [5].

rect answers could occur more frequently in those passages and usually share some commonalities, while incorrect answers are usually different from one another. The example in Table 1 demonstrates this phenomenon. We can see that the answer candidates extracted from the last four passages are all valid answers to the question and they are semantically similar to each other, while the answer candidates from the other two passages are incorrect and there is no supportive information from other passages. As human beings usually compare the answer candidates from different sources to deduce the final answer, we hope that MRC model can also benefit from the cross-passage answer verification process.

The overall framework of our model is demonstrated in Figure 1 , which consists of three modules. First, we follow the boundary-based MRC models (Seo et al., 2016; Wang and Jiang, 2016) to find an answer candidate for each passage by identifying the start and end position of the answer (Figure 2). Second, we model the meanings of the answer candidates extracted from those passages and use the content scores to measure the quality of the candidates from a second perspective. Third, we conduct the answer verification by enabling each answer candidate to attend to the other candidates based on their representations. We hope that the answer candidates can collect supportive information from each other according to their semantic similarities and further decide whether each candidate is correct or not.

Therefore, the final answer is determined by three factors: the boundary, the content and the cross-passage answer verification. The three steps are modeled using different modules, which can be jointly trained in our end-to-end framework.

We conduct extensive experiments on the MS-MARCO (Nguyen et al., 2016) and DuReader (He et al., 2017) datasets. The results show that our answer verification MRC model outperforms the baseline models by a large margin and achieves the state-of-the-art performance on both datasets.

2 Our Approach

Figure 1 gives an overview of our multi-passage MRC model which is mainly composed of three modules including answer boundary prediction, answer content modeling and answer verification. First of all, we need to model the question and passages. Following Seo et al. (2016), we compute the question-aware representation for each passage (Section 2.1). Based on this representation, we employ a Pointer Network (Vinyals et al., 2015) to predict the start and end position of the answer in the module of answer boundary prediction (Section 2.2). At the same time, with the answer content model (Section 2.3), we estimate whether each word should be included in the answer and thus obtain the answer representations. Next, in the answer verification module (Section 2.4), each answer candidate can attend to the other answer candidates to collect supportive information and we compute one score for each candidate
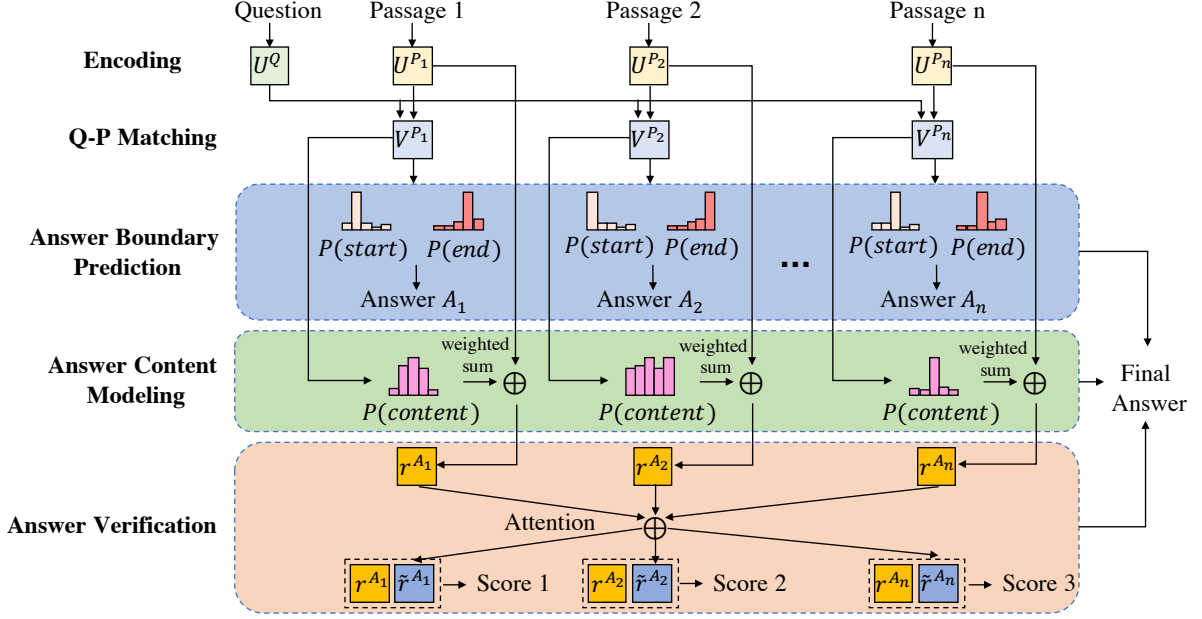
Figure 1: Overview of our method for multi-passage machine reading comprehension

to indicate whether it is correct or not according to the verification. The final answer is determined by not only the boundary but also the answer content and its verification score (Section 2.5).

2.1 Question and Passage Modeling

Given a question $\mathbf{Q}$ and a set of passages $\{\mathbf{P}_i\}$ retrieved by search engines, our task is to find the best concise answer to the question. First, we formally present the details of modeling the question and passages.

Encoding We first map each word into the vector space by concatenating its word embedding and sum of its character embeddings. Then we employ bi-directional LSTMs (BiLSTM) to encode the question $\mathbf{Q}$ and passages $\{\mathbf{P}_i\}$ as follows:

$$\mathbf{u}_t^Q = \text{BiLSTM}_Q(\mathbf{u}_{t-1}^Q, [\mathbf{e}_t^Q, \mathbf{c}_t^Q]) \qquad (1)$$

$$\mathbf{u}_t^{P_i} = \text{BiLSTM}_P(\mathbf{u}_{t-1}^{P_i}, [\mathbf{e}_t^{P_i}, \mathbf{c}_t^{P_i}]) \qquad (2)$$

where $\mathbf{e}_t^Q$, $\mathbf{c}_t^Q$, $\mathbf{e}_t^{P_i}$, $\mathbf{c}_t^{P_i}$ are the word-level and character-level embeddings of the tth word. $\mathbf{u}_t^Q$ and $\mathbf{u}_t^{P_i}$ are the encoding vectors of the tth words in $\mathbf{Q}$ and $\mathbf{P}_i$ respectively. Unlike previous work (Wang et al., 2017c) that simply concatenates all the passages, we process the passages independently at the encoding and matching steps.

Q-P Matching One essential step in MRC is to match the question with passages so that important information can be highlighted. We use the

Attention Flow Layer (Seo et al., 2016) to conduct the Q-P matching in two directions. The similarity matrix $\mathbf{S} \in \mathbb{R}^{|\mathbf{Q}| \times |\mathbf{P}_i|}$ between the question and passage i is changed to a simpler version, where the similarity between the tth word in the question and the kth word in passage i is computed as:

$$\mathbf{S}_{t,k} = \mathbf{u}_t^{Q\mathsf{T}} \cdot \mathbf{u}_k^{P_i} \qquad (3)$$

Then the context-to-question attention and question-to-context attention is applied strictly following Seo et al. (2016) to obtain the question-aware passage representation $\{\tilde{\mathbf{u}}_t^{P_i}\}$. We do not give the details here due to space limitation. Next, another BiLSTM is applied in order to fuse the contextual information and get the new representation for each word in the passage, which is regarded as the match output:

$$\mathbf{v}_t^{P_i} = \text{BiLSTM}_M(\mathbf{v}_{t-1}^{P_i}, \tilde{\mathbf{u}}_t^{P_i}) \qquad (4)$$

Based on the passage representations, we introduce the three main modules of our model.

2.2 Answer Boundary Prediction

To extract the answer span from passages, mainstream studies try to locate the boundary of the answer, which is called boundary model. Following (Wang and Jiang, 2016), we employ Pointer Network (Vinyals et al., 2015) to compute the probability of each word to be the start or end position

of the span:

$$g_k^t = \mathbf{w}_1^{a\top} \tanh(\mathbf{W}_2^a [\mathbf{v}_k^P, \mathbf{h}_{t-1}^a]) \qquad (5)$$

$$\alpha_k^t = \exp(g_k^t) / \sum\nolimits_{j=1}^{|\mathbf{P}|} \exp(g_j^t) \qquad (6)$$

$$\mathbf{c}_t = \sum\nolimits_{k=1}^{|\mathbf{P}|} \alpha_k^t \mathbf{v}_k^P \qquad (7)$$

$$\mathbf{h}_t^a = \text{LSTM}(\mathbf{h}_{t-1}^a, \mathbf{c}_t) \qquad (8)$$

By utilizing the attention weights, the probability of the kth word in the passage to be the start and end position of the answer is obtained as α_k^1 and α_k^2. It should be noted that the pointer network is applied to the concatenation of all passages, which is denoted as P so that the probabilities are comparable across passages. This boundary model can be trained by minimizing the negative log probabilities of the true start and end indices:

$$\mathcal{L}_{boundary} = -\frac{1}{N} \sum_{i=1}^{N} (\log \alpha_{y_i^1}^1 + \log \alpha_{y_i^2}^2) \qquad (9)$$

where N is the number of samples in the dataset and y_i^1, y_i^2 are the gold start and end positions.

2.3 Answer Content Modeling

Previous work employs the boundary model to find the text span with the maximum boundary score as the final answer. However, in our context, besides locating the answer candidates, we also need to model their meanings in order to conduct the verification. An intuitive method is to compute the representation of the answer candidates separately after extracting them, but it could be hard to train such model end-to-end. Here, we propose a novel method that can obtain the representation of the answer candidates based on probabilities.

Specifically, we change the output layer of the classic MRC model. Besides predicting the boundary probabilities for the words in the passages, we also predict whether each word should be included in the content of the answer. The content probability of the kth word is computed as:

$$p_k^c = \text{sigmoid}(\mathbf{w}_1^{c\top} \text{ReLU}(\mathbf{W}_2^c \mathbf{v}_k^{P_i})) \qquad (10)$$

Training this content model is also quite intuitive. We transform the boundary labels into a continuous segment, which means the words within the answer span will be labeled as 1 and other words will be labeled as 0. In this way, we define

the loss function as the averaged cross entropy:

$$\mathcal{L}_{content} = -\frac{1}{N} \frac{1}{|\mathbf{P}|} \sum_{i=1}^{N} \sum_{j=1}^{|\mathbf{P}|} [y_k^c \log p_k^c \qquad (11)$$
$$+ (1 - y_k^c) \log(1 - p_k^c)]$$

The content probabilities provide another view to measure the quality of the answer in addition to the boundary. Moreover, with these probabilities, we can represent the answer from passage i as a weighted sum of all the word embeddings in this passage:

$$\mathbf{r}^{A_i} = \frac{1}{|\mathbf{P}_i|} \sum\nolimits_{k=1}^{|\mathbf{P}_i|} p_k^c [\mathbf{e}_k^{P_i}, \mathbf{c}_k^{P_i}] \qquad (12)$$

2.4 Cross-Passage Answer Verification

The boundary model and the content model focus on extracting and modeling the answer within a single passage respectively, with little consideration of the cross-passage information. However, as is discussed in Section 1, there could be multiple answer candidates from different passages and some of them may mislead the MRC model to make an incorrect prediction. It's necessary to aggregate the information from different passages and choose the best one from those candidates. Therefore, we propose a method to enable the answer candidates to exchange information and verify each other through the cross-passage answer verification process.

Given the representation of the answer candidates from all passages $\{\mathbf{r}^{A_i}\}$, each answer candidate then attends to other candidates to collect supportive information via attention mechanism:

$$s_{i,j} = \begin{cases} 0, & \text{if } i = j, \\ \mathbf{r}^{A_i\top} \cdot \mathbf{r}^{A_j}, & \text{otherwise} \end{cases} \qquad (13)$$

$$\alpha_{i,j} = \exp(s_{i,j}) / \sum\nolimits_{k=1}^{n} \exp(s_{i,k}) \qquad (14)$$

$$\tilde{\mathbf{r}}^{A_i} = \sum\nolimits_{j=1}^{n} \alpha_{i,j} \mathbf{r}^{A_j} \qquad (15)$$

Here $\tilde{\mathbf{r}}^{A_i}$ is the collected verification information from other passages based on the attention weights. Then we pass it together with the original representation $\mathbf{r}^{A_i}$ to a fully connected layer:

$$g_i^v = \mathbf{w}^{v\top} [\mathbf{r}^{A_i}, \tilde{\mathbf{r}}^{A_i}, \mathbf{r}^{A_i} \odot \tilde{\mathbf{r}}^{A_i}] \qquad (16)$$

We further normalize these scores over all passages to get the verification score for answer candidate A_i:

$$p_i^v = \exp(g_i^v) / \sum\nolimits_{j=1}^{n} \exp(g_j^v) \qquad (17)$$

In order to train this verification model, we take the answer from the gold passage as the gold answer. And the loss function can be formulated as the negative log probability of the correct answer:

$$\mathcal{L}_{verify} = -\frac{1}{N} \sum_{i=1}^{N} \log p_{y_i^v}^v \qquad (18)$$

where y_i^v is the index of the correct answer in all the answer candidates of the ith instance .

2.5 Joint Training and Prediction

As is described above, we define three objectives for the reading comprehension model over multiple passages: 1. finding the boundary of the answer; 2. predicting whether each word should be included in the content; 3. selecting the best answer via cross-passage answer verification. According to our design, these three tasks can share the same embedding, encoding and matching layers. Therefore, we propose to train them together as multi-task learning (Ruder, 2017). The joint objective function is formulated as follows:

$$\mathcal{L} = \mathcal{L}_{boundary} + \beta_1 \mathcal{L}_{content} + \beta_2 \mathcal{L}_{verify} \quad (19)$$

where β_1 and β_2 are two hyper-parameters that control the weights of those tasks.

When predicting the final answer, we take the boundary score, content score and verification score into consideration. We first extract the answer candidate A_i that has the maximum boundary score from each passage i. This boundary score is computed as the product of the start and end probability of the answer span. Then for each answer candidate A_i, we average the content probabilities of all its words as the content score of A_i. And we can also predict the verification score for A_i using the verification model. Therefore, the final answer can be selected from all the answer candidates according to the product of these three scores.

3 Experiments

To verify the effectiveness of our model on multi-passage machine reading comprehension, we conduct experiments on the MS-MARCO (Nguyen et al., 2016) and DuReader (He et al., 2017) datasets. Our method achieves the state-of-the-art performance on both datasets.

3.1 Datasets

We choose the MS-MARCO and DuReader datasets to test our method, since both of them are

	MS-MARCO	DuReader
Multiple Answers	9.93%	67.28%
Multiple Spans	40.00%	56.38%

Table 2: Percentage of questions that have multiple valid answers or answer spans

designed from real-world search engines and involve a large number of passages retrieved from the web. One difference of these two datasets is that MS-MARCO mainly focuses on the English web data, while DuReader is designed for Chinese MRC. This diversity is expected to reflect the generality of our method. In terms of the data size, MS-MARCO contains 102023 questions, each of which is paired up with approximately 10 passages for reading comprehension. As for DuReader, it keeps the top-5 search results for each question and there are totally 201574 questions.

One prerequisite for answer verification is that there should be multiple correct answers so that they can verify each other. Both the MS-MARCO and DuReader datasets require the human annotators to generate multiple answers if possible. Table 2 shows the proportion of questions that have multiple answers. However, the same answer that occurs many times is treated as one single answer here. Therefore, we also report the proportion of questions that have multiple answer spans to match with the human-generated answers. A span is taken as valid if it can achieve F1 score larger than 0.7 compared with any reference answer. From these statistics, we can see that the phenomenon of multiple answers is quite common for both MS-MARCO and DuReader. These answers will provide strong signals for answer verification if we can leverage them properly.

3.2 Implementation Details

For MS-MARCO, we preprocess the corpus with the reversible tokenizer from Stanford CoreNLP (Manning et al., 2014) and we choose the span that achieves the highest ROUGE-L score with the reference answers as the gold span for training. We employ the 300-D pre-trained Glove embeddings (Pennington et al., 2014) and keep it fixed during training. The character embeddings are randomly initialized with its dimension as 30. For DuReader, we follow the preprocessing described in He et al. (2017).

We tune the hyper-parameters according to the

Model	ROUGE-L	BLEU-1
FastQA_Ext (Weissenborn et al., 2017)	33.67	33.93
Prediction (Wang and Jiang, 2016)	37.33	40.72
ReasoNet (Shen et al., 2017)	38.81	39.86
R-Net (Wang et al., 2017c)	42.89	42.22
S-Net (Tan et al., 2017)	45.23	43.78
Our Model	**46.15**	**44.47**
S-Net (Ensemble)	46.65	44.78
Our Model (Ensemble)	**46.66**	**45.41**
Human	47	46

Table 3: Performance of our method and competing models on the MS-MARCO test set

validation performance on the MS-MARCO development set. The hidden size is set to be 150 and we apply $L2$ regularization with its weight as 0.0003. The task weights β_1, β_2 are both set to be 0.5. To train our model, we employ the Adam algorithm (Kingma and Ba, 2014) with the initial learning rate as 0.0004 and the mini-batch size as 32. Exponential moving average is applied on all trainable variables with a decay rate 0.9999.

Two simple yet effective technologies are employed to improve the final performance on these two datasets respectively. For MS-MARCO, approximately 8% questions have the answers as Yes or No, which usually cannot be solved by extractive approach (Tan et al., 2017). We address this problem by training a simple Yes/No classifier for those questions with certain patterns (e.g., starting with "is"). Concretely, we simply change the output layer of the basic boundary model so that it can predict whether the answer is "Yes" or "No". For DuReader, the retrieved document usually contains a large number of paragraphs that cannot be fed into MRC models directly (He et al., 2017). The original paper employs a simple simple heuristic strategy to select a representative paragraph for each document, while we train a paragraph ranking model for this. We will demonstrate the effects of these two technologies later.

3.3 Results on MS-MARCO

Table 3 shows the results of our system and other state-of-the-art models on the MS-MARCO test set. We adopt the official evaluation metrics, including ROUGE-L (Lin, 2004) and BLEU-1 (Papineni et al., 2002). As we can see, for both metrics, our single model outperforms all the other competing models with an evident margin, which is extremely hard considering the near-human per-

Model	BLEU-4	ROUGE-L
Match-LSTM	31.8	39.0
BiDAF	31.9	39.2
PR + BiDAF	37.55	41.81
Our Model	**40.97**	**44.18**
Human	56.1	57.4

Table 4: Performance on the DuReader test set

Model	ROUGE-L	Δ
Complete Model	**45.65**	**-**
Answer Verification	44.38	-1.27
Content Modeling	44.27	-1.38
Joint Training	44.12	-1.53
YesNo Classification	41.87	-3.78
Boundary Baseline	**38.95**	**-6.70**

Table 5: Ablation study on MS-MARCO development set

formance. If we ensemble the models trained with different random seeds and hyper-parameters, the results can be further improved and outperform the ensemble model in Tan et al. (2017), especially in terms of the BLEU-1.

3.4 Results on DuReader

The results of our model and several baseline systems on the test set of DuReader are shown in Table 4. The BiDAF and Match-LSTM models are provided as two baseline systems (He et al., 2017). Based on BiDAF, as is described in Section 3.2, we tried a new paragraph selection strategy by employing a paragraph ranking (PR) model. We can see that this paragraph ranking can boost the BiDAF baseline significantly. Finally, we implement our system based on this new strategy, and our system (single model) achieves further improvement by a large margin.

Question: What is the difference between a mixed and pure culture	Scores		
Answer Candidates:	Boundary	Content	Verification
[1] A culture is a society's total way of living and a society is a group . . .	1.0×10^{-2}	1.0×10^{-1}	1.1×10^{-1}
[2] The mixed economy is a balance between socialism and capitalism.	1.0×10^{-4}	4.0×10^{-2}	3.2×10^{-2}
[3] A pure culture is one in which only one kind of microbial species is . . .	5.5×10^{-3}	7.7×10^{-2}	1.2×10^{-1}
[4] A pure culture comprises a single species or strains. A mixed . . .	2.7×10^{-3}	8.1×10^{-2}	1.3×10^{-1}
[5] A pure culture is a culture consisting of only one strain.	5.8×10^{-4}	7.9×10^{-2}	5.1×10^{-2}
[6] A pure culture is one in which only one kind of microbial species . . .	5.8×10^{-3}	9.1×10^{-2}	$\mathbf{2.7 \times 10^{-1}}$
. . . . . .		. . . . . .	

Table 6: Scores predicted by our model for the answer candidates shown in Table 1. Although the candidate [1] gets high boundary and content scores, the correct answer [6] is preferred by the verification model and is chosen as the final answer.

4 Analysis and Discussion

4.1 Ablation Study

To get better insight into our system, we conduct in-depth ablation study on the development set of MS-MARCO, which is shown in Table 5. Following Tan et al. (2017), we mainly focus on the ROUGE-L score that is averaged case by case.

We first evaluate the answer verification by ablating the cross-passage verification model so that the verification loss and verification score will not be used during training and testing. Then we remove the content model in order to test the necessity of modeling the content of the answer. Since we don't have the content scores, we use the boundary probabilities instead to compute the answer representation for verification. Next, to show the benefits of joint training, we train the boundary model separately from the other two models. Finally, we remove the yes/no classification in order to show the real improvement of our end-to-end model compared with the baseline method that predicts the answer with only the boundary model.

From Table 5, we can see that the answer verification makes a great contribution to the overall improvement, which confirms our hypothesis that cross-passage answer verification is useful for the multi-passage MRC. For the ablation of the content model, we analyze that it will not only affect the content score itself, but also violate the verification model since the content probabilities are necessary for the answer representation, which will be further analyzed in Section 4.3. Another discovery is that jointly training the three models can provide great benefits, which shows that the three tasks are actually closely related and can boost each other with shared representations at bottom layers. At last, comparing our method with the baseline, we achieve an improvement of nearly

3 points without the yes/no classification. This significant improvement proves the effectiveness of our approach.

4.2 Case Study

To demonstrate how each module of our model takes effect when predicting the final answer, we conduct a case study in Table 6 with the same example that we discussed in Section 1. For each answer candidate, we list three scores predicted by the boundary model, content model and verification model respectively.

On the one hand, we can see that these three scores generally have some relevance. For example, the second candidate is given lowest scores by all the three models. We analyze that this is because the models share the same encoding and matching layers at bottom level and this relevance guarantees that the content and verification models will not violate the boundary model too much. On the other hand, we also see that the verification score can really make a difference here when the boundary model makes an incorrect decision among the confusing answer candidates ([1], [3], [4], [6]). Besides, as we expected, the verification model tends to give higher scores for those answers that have semantic commonality with each other ([3], [4], [6]), which are all valid answers in this case. By multiplying the three scores, our model finally predicts the answer correctly.

4.3 Necessity of the Content Model

In our model, we compute the answer representation based on the content probabilities predicted by a separate content model instead of directly using the boundary probabilities. We argue that this content model is necessary for our answer verification process. Figure 2 plots the predicted content probabilities as well as the boundary probabilities
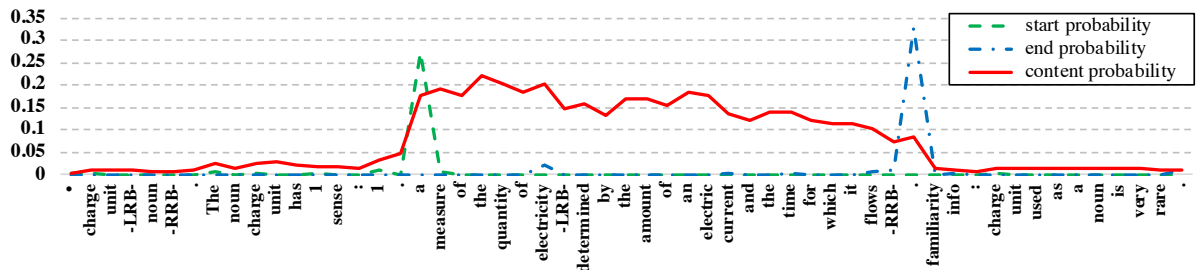
Figure 2: The boundary probabilities and content probabilities for the words in a passage

for a passage. We can see that the boundary and content probabilities capture different aspects of the answer. Since answer candidates usually have similar boundary words, if we compute the answer representation based on the boundary probabilities, it's difficult to model the real difference among different answer candidates. On the contrary, with the content probabilities, we pay more attention to the content part of the answer, which can provide more distinguishable information for verifying the correct answer. Furthermore, the content probabilities can also adjust the weights of the words within the answer span so that unimportant words (e.g. "and" and ".") get lower weights in the final answer representation. We believe that this refined representation is also good for the answer verification process.

5 Related Work

Machine reading comprehension made rapid progress in recent years, especially for single-passage MRC task, such as SQuAD (Rajpurkar et al., 2016). Mainstream studies (Seo et al., 2016; Wang and Jiang, 2016; Xiong et al., 2016) treat reading comprehension as extracting answer span from the given passage, which is usually achieved by predicting the start and end position of the answer. We implement our boundary model similarly by employing the boundary-based pointer network (Wang and Jiang, 2016). Another inspiring work is from Wang et al. (2017c), where the authors propose to match the passage against itself so that the representation can aggregate evidence from the whole passage. Our verification model adopts a similar idea. However, we collect information across passages and our attention is based on the answer representation, which is much more efficient than attention over all passages. For the model training, Xiong et al. (2017) argues that the boundary loss encourages exact answers at the

cost of penalizing overlapping answers. Therefore they propose a mixed objective that incorporates rewards derived from word overlap. Our joint training approach has a similar function. By taking the content and verification loss into consideration, our model will give less loss for overlapping answers than those unmatched answers, and our loss function is totally differentiable.

Recently, we also see emerging interests in multi-passage MRC from both the academic (Dunn et al., 2017; Joshi et al., 2017) and industrial community (Nguyen et al., 2016; He et al., 2017). Early studies (Shen et al., 2017; Wang et al., 2017c) usually concat those passages and employ the same models designed for single-passage MRC. However, more and more latest studies start to design specific methods that can read multiple passages more effectively. In the aspect of passage selection, Wang et al. (2017a) introduced a pipelined approach that rank the passages first and then read the selected passages for answering questions. Tan et al. (2017) treats the passage ranking as an auxiliary task that can be trained jointly with the reading comprehension model. Actually, the target of our answer verification is very similar to that of the passage selection, while we pay more attention to the answer content and the answer verification process. Speaking of the answer verification, Wang et al. (2017b) has a similar motivation to ours. They attempt to aggregate the evidence from different passages and choose the final answer from n-best candidates. However, they implement their idea as a separate reranking step after reading comprehension, while our answer verification is a component of the whole model that can be trained end-to-end.

6 Conclusion

In this paper, we propose an end-to-end framework to tackle the multi-passage MRC task . We

creatively design three different modules in our model, which can find the answer boundary, model the answer content and conduct cross-passage answer verification respectively. All these three modules can be trained with different forms of the answer labels and training them jointly can provide further improvement. The experimental results demonstrate that our model outperforms the baseline models by a large margin and achieves the state-of-the-art performance on two challenging datasets, both of which are designed for MRC on real web data.

Acknowledgments

References

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the cnn/daily mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*.

Matthew Dunn, Levent Sagun, Mike Higgins, V. Ugur Güney, Volkan Cirik, and Kyunghyun Cho. 2017. Searchqa: A new q&a dataset augmented with context from a search engine. *arXiv preprint arXiv:1704.05179* .

Wei He, Kai Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, Tian Wu, and Haifeng Wang. 2017. Dureader: a chinese machine reading comprehension dataset from real-world applications. *arXiv preprint arXiv:1711.05073* .

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

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

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing, EMNLP 2017, Copenhagen, Denmark, September 9-11, 2017*. pages 2021–2031.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. volume 1, pages 1601–1611.

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

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* .

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

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

Boyuan Pan, Hao Li, Zhou Zhao, Bin Cao, Deng Cai, and Xiaofei He. 2017. Memen: Multi-layer embedding with memory networks for machine comprehension. *arXiv preprint arXiv:1707.09098* .

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA.*. pages 311–318.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*. pages 1532–1543.

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

Sebastian Ruder. 2017. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098* .

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

Yelong Shen, Po-Sen Huang, Jianfeng Gao, and Weizhu Chen. 2017. Reasonet: Learning to stop reading in machine comprehension. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017*. pages 1047–1055.

Chuanqi Tan, Furu Wei, Nan Yang, Weifeng Lv, and Ming Zhou. 2017. S-net: From answer extraction to answer generation for machine reading comprehension. *arXiv preprint arXiv:1706.04815* .

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. 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*. pages 2692–2700.

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

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerald Tesauro, Bowen Zhou, and Jing Jiang. 2017a. R$ˆ3$: Reinforced reader-ranker for open-domain question answering. *arXiv preprint arXiv:1709.00023* .

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2017b. Evidence aggregation for answer re-ranking in open-domain question answering. *arXiv preprint arXiv:1711.05116* .

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017c. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017), Vancouver, Canada, August 3-4, 2017*. pages 271–280.

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

Caiming Xiong, Victor Zhong, and Richard Socher. 2017. DCN+: mixed objective and deep residual coattention for question answering. *arXiv preprint arXiv:1711.00106* .