A Multi-answer Multi-task Framework for Real-world Machine Reading Comprehension

Jiahua Liu[1,2], **Wan Wei**[2], **Maosong Sun**[1], **Hao Chen**[2], **Yantao Du**[2], **Dekang Lin**[2*]
[1]State Key Laboratory of Intelligent Technology and Systems,
Tsinghua National Laboratory for Information Science and Technology,
Department of Computer Science and Technology, Tsinghua University, Beijing, China
[2]Naturali Ltd, Beijing, China

Abstract

The task of machine reading comprehension (MRC) has evolved from answering simple questions from well-edited text to answering real questions from users out of web data. In the real-world setting, full-body text from multiple relevant documents in the top search results are provided as context for questions from user queries, including not only questions with a single, short, and factual answer, but also questions about reasons, procedures, and opinions. In this case, multiple answers could be equally valid for a single question and each answer may occur multiple times in the context, which should be taken into consideration when we build MRC system. We propose a multi-answer multi-task framework, in which different loss functions are used for multiple reference answers. Minimum Risk Training is applied to solve the multi-occurrence problem of a single answer. Combined with a simple heuristic passage extraction strategy for overlong documents, our model increases the ROUGE-L score on the DuReader dataset from 44.18, the previous state-of-the-art, to 51.09.

1 Introduction

Machine reading comprehension (MRC) or question answering (QA) has been a long-standing goal in Natural Language Processing. There is a surge of interest in this area due to new end-to-end modeling techniques and the release of several large-scale, open-domain datasets.

In earlier datasets (Hermann et al., 2015; Hill et al., 2016; Yang et al., 2015; Rajpurkar et al., 2016), the questions did not arise from actual end users. Instead, they were constructed in cloze style or created by crowdworkers given a short passage from well-edited sources such as Wikipedia and CNN/Daily Mail. As a consequence, the questions are usually well-formed and about simple facts, and the answers are guaranteed to exist as short spans in the given candidate passages.

In MS-MARCO (Nguyen et al., 2016), the questions were sampled from actual search queries, which may have typos and may not be phrased as questions.[1] Multiple short passages, which might have the answer to the query, were extracted from webpages by a separate information retrieval system.

He et al. (2017) made the DuReader dataset a more realistic reflection of the real-world problem by including not only questions with relatively short and factual answers, but also questions about complex descriptions, procedures, opinions, etc. which may have multiple, much longer answers, or no answer at all. Furthermore, full-body text from webpages listed in top search results are directly provided as context. These documents tend to be much noiser than Wikipedia and CNN. They are much longer (5 times longer than those in MS-MARCO on average) and contain many paragraphs that are irrelevant to the query.

New problems arise as we now consider the task of machine reading comprehension in a much more challenging real-world setting. First, multiple valid answers to a single question are not only possible but quite common. Figure 1 shows some examples of questions with multiple answers from the DuReader dataset. There could be multiple ways to perform the same task (Question 1), multiple opinions about the same subject (Question 2), or multiple explanations for the same observation (Question 3). However, few works have been done with multiple answers in machine reading comprehension. To address this problem, we propose a multi-answer multi-task scheme which incorporates multiple reference answers in the objective

*Corresponding author: D. Lin (lindek@naturali.io).

[1]We use these two terms, question and query, interchangeably in the following content

Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2109–2118
Brussels, Belgium, October 31 - November 4, 2018.

Question 1:
word字间距
character spacing in Word
Answers:
1. 选择要缩进的内容，点击右键选择字体，打开“字体”对话框，选择“字符间距”选项卡，勾选“为字体调整字间距”并输入数字，最后单击“确定”按钮。
2. 选中你要设置字符间距的文字后，单击鼠标右键，选择字体选项，在切换到字符间距。
3. 点击上部菜单栏上“格式”→“字体”→“字符间距”，调节间距的磅值即可。
1. Select the text you want to indent, and right-click on it to select Font, open the Font dialog box, and then select the Character Spacing tab, select the Kerning for fonts check box, and enter a number, at last click OK.
2. Select the text you want to set character spacing, and right-click on it to select Font, and then switch to Character Spacing.
3. click Format from the upper menu bar, select Font, select Character Spacing, and then change the point size.

Question 2:

龙珠传奇好看吗	Is LONGZHU worth watching?
Answers:	
1. 好看的、节奏还可以、萌萌的甜甜的。	1. It is, the pace is good and it's adorable and sweet.
2. 不好看，个人不喜欢那种类型的电视剧。	2. It’s not, personally I don’t like that kind of TV series.
3. 还行，不纠结历史问题的话可追。	3. It’s ok, it’s fine to follow if you don’t care about the historical accuracy.

Question 3:

响一声就说正在通话中	Got “User Busy” message after one ring
Answers:	
1. 别人对你设置了黑名单。	1. Your number is in that person’s blacklist.
2. 别人挂了你的电话。	2. The person you called hung up on you.

Figure 1: Examples of questions with multiple answers from the DuReader dataset

function (but still predicts a single answer in decoding time). We propose three different kinds of multi-answer loss functions and compare their performance through experiment.

Another problem is the multiple occurrences of the same answer. As rich context is provided for a single question, the same answer could occur more than one time in different passages, or even at different places of the same passage. In this case, using only one gold span for the answer could be problematic, as the model is forced to choose one span over others that contain the same content. To solve this problem, we propose to apply Minimum Risk Training (MRT), which uses the expected metric as the loss and gives reward to all spans that are similar with the gold answer.

In this paper, we present a multi-answer, multi-task objective function to train an end-to-end MRC/QA system. We experiment with various alternatives on the DuReader dataset and show that our model out-performs other competing systems and increases the state-of-the-art ROUGE-L score by about 7 points.

2 Related Work

Various datasets have been released in recent years, which fuel the research for reading comprehension and question answering. The CNN/Daily-Mail dataset (Hermann et al., 2015) and the Children’s Book Test (Hill et al., 2016) evaluate comprehension by filling in missing words from well-edited texts. SQuAD (Rajpurkar et al., 2016) is one of the most popular datasets for reading comprehension, where a span in a Wikipedia passage is to be extracted to answer questions generated by annotators. The WikiQA (Yang et al., 2015) is another dataset from Wikipedia, where one single sentence is to be selected to answer questions from search engine logs. Different from the above datasets, the MS-MARCO dataset (Nguyen et al., 2016) was built in a real-world setting. The questions were real anonymized Bing queries and multiple passages are extracted from related web pages by a separate system. The DuReader (He et al., 2017) is a Chinese dataset, similarly constructed from user queries as MS MACRCO, but in a more realistic setting using Baidu Web Search and Baidu Answers (Zhidao) data. While a small proportion of questions were labeled with multiple answers in MS MARCO (9.93%), more than half of the DuReader queries were annotated with multiple answers, which provides the perfect setup for our work.

Great effort has been put into the development of sophisticated neural models for machine reading comprehension. The attention mechanism was first introduced by Hermann et al. (2015) into reading comprehension and soon became the dom-

inating model. Wang and Jiang (2017) proposed to solve machine comprehension using Match-LSTM and answer pointer. Seo et al. (2017) and Xiong et al. (2017) applied different ways to match the question and the context with bi-directional attention. Hu et al. (2017) used iterative aligner to match the question and the passage with feature-rich encoder. Cui et al. (2017) employed one more layer of attention over the bi-directional attention mechanism. Wang et al. (2017) applied a self-matching mechanism to aggregate evidence from the context. Tan et al. (2018) proposed to generate answer from extracted answer span. Yu et al. (2018) proposed to use convolution with self-attention instead of recurrent models in reading comprehension.

Recently there are some emerging works starting to touch the reading comprehension task from the answer side. Wang et al. (2018a) proposed to use evidence aggregation to re-rank answer candidates extracted from different passages, and Wang et al. (2018b) proposed Cross-Passage Answer Verification model for the same purpose. Neither of them involved multiple answers as in this work.

Minimum Risk Training (MRT) has been widely used in various tasks in NLP. Shen et al. (2016) introduced MRT into Neural Machine Translation, and Ayana et al. (2016) applied it in Text Summarization.

3 Our Approach

In this section we describe in details the architecture of our model which is depicted in Figure 2.

3.1 Passage Extraction

Unlike most other datasets where the source of answers is a short passage with a few hundred words, the DuReader dataset provides up to 5 full documents, which may contain up to 100K words. This incurs exorbitant demand on memory and training time. To deal with this issue, previous approaches select a single representative paragraph for each document, on which the answer extraction is performed. The original paper of DuReader (He et al., 2017) employed a simple heuristic strategy, and Wang et al. (2018b) trained a paragraph ranking model, while Clark and Gardner (2017) applied TF-IDF based method for the TriviaQA dataset (Joshi et al., 2017) which was in a similar situation. However, answers could come from more than one paragraph. We apply a simple yet effective method to extract contents from multiple paragraphs of the document, aiming to include as much information for the answer extraction as possible.

We concatenate the title and the whole document as the passage if it is shorter than a predefined maximum length. If not, we employ passage extraction in the following way:

- The title of the document is extracted. Whether a document is relevant to the question could be easily seen from the title.
- We compute BLEU-4 score of each paragraph relative to the question, and select the one that appears first in the document among paragraphs with top-k scores.
- We extract the full body of this selected paragraph and the next paragraph.
- For each of the following paragraphs, the first sentence is extracted as it probably contains the main point.
- We concatenate all the extracted contents to form the extracted passage, and it is truncated to the maximum length if it is longer than the predefined value.

We apply our model on the basis of the extracted passages.

3.2 Representation of Word

Given a word sequence of question $Q = \{w_t^q\}_{t=1}^m$, and a word sequence of extracted passage $P = \{w_t^p\}_{t=1}^n$, we combine different useful information to form the representation of each question word w_t^q and passage word w_t^p:

- Word-level embedding: each word w in the question and passage is mapped to its corresponding n-dimensional embedding we.
- POS tag embedding: we use a POS tagger to tag each word in the question and passage. Each POS tag is mapped to a m-dimensional embedding pe.
- Word-in-question feature: following Chen et al. (2017) and Weissenborn et al. (2017), we use one additional binary feature wiq for each passage word, indicating whether this word occurs in the question.

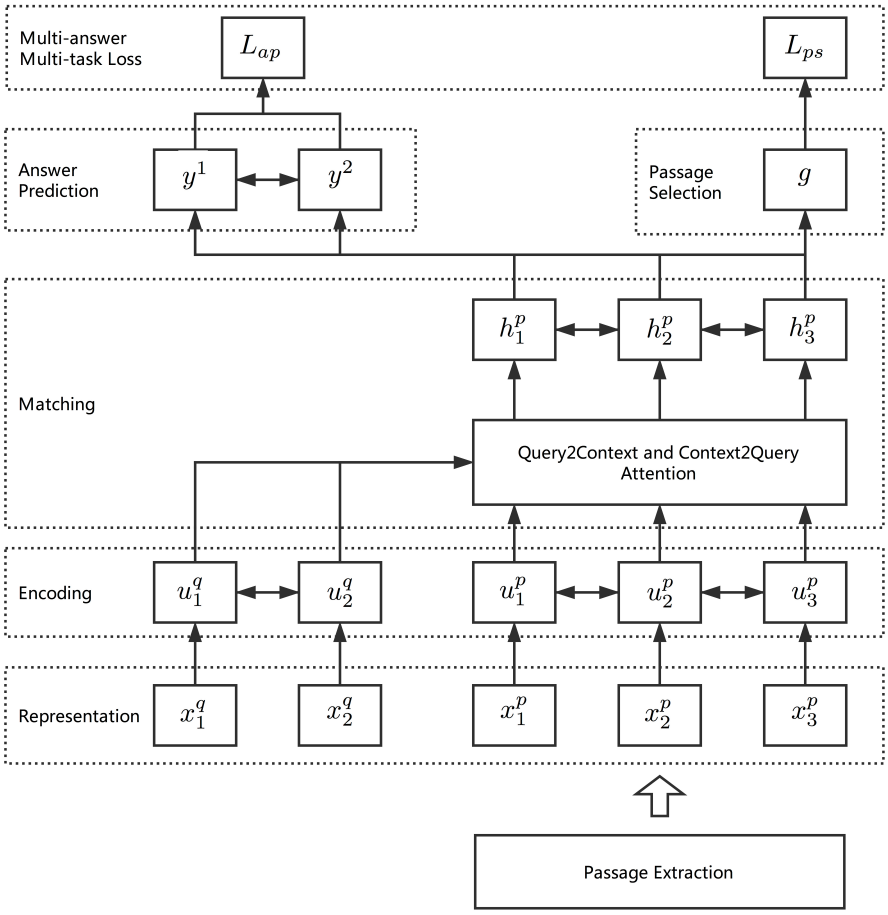

Figure 2: Model Architecture

Each question word is represented as the concatenation of the word embedding we, and the POS tag embedding pe, denoted as $x^q = [we; pe]$. Each passage word is additionally concatenated with the word-in-question feature wiq: $x^p = [we; pe; wiq]$.

It should be noted that, character-level embedding is an important part of word representation in English MRC models (Seo et al., 2017; Weissenborn et al., 2017; Wang et al., 2017; Tan et al., 2018). Character sequence would give information which helps to relieve the OOV problem, as many English words share the same stem and differ only in prefix or suffix. However, this is not the case in Chinese, and we observe no significant improvement incorporating character-level embedding into our system.

3.3 Encoding Layer

Following previous work, we use a bi-directional LSTM to obtain contextual encoding for each word in the question and passage respectively:

$$u_i^q = BiLSTM(u_{i-1}^q, x_i^q) \quad (1)$$

$$u_j^p = BiLSTM(u_{j-1}^p, x_j^p) \quad (2)$$

3.4 Match Layer

To fuse question encoding and passage encoding, we adopt the Attention Flow Layer (Seo et al., 2017) with a simpler similarity function. The similarity score between the contextual encoding for a query word u_i^q and that for a passage word u_j^p is defined as:

$$s_{ij} = u_i^{qT} \cdot u_j^p \quad (3)$$

The context-to-query attention vectors c_j^p are computed from the similarity scores:

$$a_{ij} = \frac{exp(s_{ij})}{\sum_{k=1}^{m} exp(s_{kj})} \quad (4)$$

$$c_j^p = \sum_{i=1}^{m} a_{ij} u_i^q \quad (5)$$

The query-to-context attention vector d^p is computed as:

$$z_j = \max_i s_{ij} \tag{6}$$

$$b_j = \frac{exp(z_j)}{\sum_{k=1}^{n} exp(z_k)} \tag{7}$$

$$d^p = \sum_{j=1}^{n} b_j u_j^p \tag{8}$$

Another BiLSTM is applied on top of them to get the question-aware passage representation:

$$h_j^p = BiLSTM(h_{j-1}^p, [u_j^p; c_j^p; u_j^p \circ c_j^p; u_j^p \circ d^p]) \tag{9}$$

3.5 Multi-answer multi-task loss function

3.5.1 Answer prediction with multi-answer

A reading comprehension model is typically trained as an extractor of an answer span from a candidate passage. In DuReader dataset, multiple reference answers are provided for a single question. For each of the reference answers, we add the span with the highest F1 score to the gold answer spans. For models considering only a single answer span (baseline model), the gold answer span is the one with the highest F1 score relative to any of the reference answers (He et al., 2017; Wang et al., 2018b).

In the boundary model with pointer network (Wang and Jiang, 2017; Wang et al., 2017; Tan et al., 2018), two probability distributions y_j^1 and y_j^2 ($j = 1 \ldots n$), which denote the probability that position j is the beginning or the end of the answer span respectively, are computed as follows:

$$s_j^t = v^T tanh(W_h^p h_j^p + W_a^P h_{j-1}^a) \tag{10}$$

$$y_j^t = \frac{exp(s_j^t)}{\sum_{k=1}^{n} exp(s_k^t)} \tag{11}$$

$$c_t = \sum_{j=1}^{n} y_j^t h_j^p \tag{12}$$

$$h_t^a = BiLSTM(h_{t-1}^a, c_t) \tag{13}$$

where $t = 1, 2$, and the initial hidden state h_0^a is generated by an attention-pooling over the question representation following Wang et al. (2017):

$$s_i = v^T tanh(W_u^q u_i^q + b) \tag{14}$$

$$a_i = \frac{exp(s_i)}{\sum_{k=1}^{m} exp(s_k)} \tag{15}$$

$$h_0^a = \sum_{i=1}^{m} a_i u_i^q \tag{16}$$

Note that all passages for the same question are concatenated in order to predict one answer span. The loss is defined as the sum of negative log probabilities of the ground truth start and end position based on the predicted distributions:

$$L = -(\log y_{start}^1 + \log y_{end}^2) \tag{17}$$

We propose three different ways to incorporate multiple answers. A simple solution is to compute the average loss for multiple answer spans:

$$L_{avg} = -\frac{1}{A} \sum_{k=1}^{A} (\log y_{start_k}^1 + \log y_{end_k}^2) \tag{18}$$

L_{avg} treats all answer spans as equally good. However, some of them may be closer to human-generated answers than others. We therefore define the weighted average loss as follow:

$$L_{wavg} = -\sum_{k=1}^{A} w_k (\log y_{start_k}^1 + \log y_{end_k}^2) \tag{19}$$

where w_k is the F-score between the answer span and the corresponding human-generated answer, normalized by the sum of the scores of each answer.

Another solution is to use the minimal value of the loss from each span:

$$L_{min} = \min_k (-(\log y_{start_k}^1 + \log y_{end_k}^2)) \tag{20}$$

Instead of predicting all answer spans, this loss will encourage the model to predict only the easiest answer span for it.

The answer span prediction loss L_{ap} is defined as the average of any of the loss functions described above over the training set.

3.5.2 Passage selection with multi-answer

Tan et al. (2018) showed that their single-answer, multi-passage MRC model benefits from using multi-task learning by adding an auxiliary loss to predict the correct passage to extract the answer from. We adapt the idea to compute passage selection loss L_{ps} in multi-answer setting.

We first apply attention-pooling over the passage representation $\{h_j^p\}_{j=1}^n$, and then calculate a matching score g for each passage:

$$s_j = v^T tanh(W_u^p h_j^p + b) \tag{21}$$

$$a_j = \frac{exp(s_j)}{\sum_{k=1}^n exp(s_k)} \tag{22}$$

$$r^p = \sum_{j=1}^n a_j h_j^p \tag{23}$$

$$g = sigmoid(v_{sp}^T r^p) \tag{24}$$

Since multiple answers are provided in the DuReader dataset, multiple passages may contain correct answers. The match score g for different passages are not in competition against one another. We therefore used pointwise sigmoid function instead of the softmax function (as in Tan et al. (2018)) in the passage selection loss L_{ps}:

$$L_{ps} = -\frac{1}{K}\sum_{k=1}^{K}(y_k \log g_k + (1-y_k)\log(1-g_k)) \tag{25}$$

where $y_k = 1$ if one of the gold span comes from this passage, $y_k = 0$ otherwise.

3.5.3 Joint training

We train our model by jointly optimizing answer span prediction loss and passage selection loss:

$$L = L_{ap} + \lambda_{ps} L_{ps} \tag{26}$$

where λ_{ps} is a hyper-parameter tuned on the dev set.

3.6 Minimum Risk Training

Minimum Risk Training (MRT) has been widely used in various tasks in NLP. The basic idea is to directly optimize the evaluation metric instead of maximizing the log likelihood of training data using Maximum Likelihood Estimation (MLE) as described above. In MRT, the object is to minimize the expected loss with respect to the posterior distribution:

$$J_{MRT}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{y_i|x_i;\theta}[\Delta(y_i, y_i^*)] \tag{27}$$

where $\Delta(y_i, y_i^*)$ is a function which indicates the difference between the predicted result y_i and the label result y_i^*.

In this work, we apply MRT to solve the problem of multi-occurrence of answer in machine reading comprehension, directly using the metric (ROUGE-L) as Δ. As an answer occurs multiple times in the context, each span in which the answer occurs will have minimum difference with the answer, and is thus given a high score by a model trained with MRT.

In machine translation and many other tasks, to compute the expected metric with respect to the posterior distribution is often intractable. Thus sampling methods are commonly used in MRT training. However, in our span extraction model, we use all spans without sampling.

Formally, the MRT loss in our model is defined as:

$$J_{MRT}(\theta) = \frac{1}{N}\sum_{i=1}^{N}\sum_{k=1}^{|P|}\sum_{l=k+1}^{|P|} y_k^1 y_l^2 \Delta(P_{k,l}, A) \tag{28}$$

As in Hu et al. (2017), we minimize the linear combination of MLE and MRT loss:

$$J(\theta) = J_{MLE}(\theta) + \lambda J_{MRT}(\theta) \tag{29}$$

where $J_{MLE}(\theta)$ refers to L in equation 26 and λ is a hyper-parameter tuned on the development set.

4 Experiment

We conduct our experiment on the DuReader dataset (He et al., 2017), where multiple passages containing full-body text are provided for each question, and over half of the questions have multiple answers.

4.1 Dataset and Evaluation Metrics

The DuReader dataset consists of 201574 questions in total, with 181574 in the training set, 10000 in the development set, and 10000 in the test set. The questions are sampled from frequently occurring queries from Baidu search engines, and the full-body text of top-5 search results from the web are provided as the context.

BLEU-4 and ROUGE-L are used in evaluation on DuReader. However, the implementations for the two metrics are quite different in the official evaluation tool. As in MS-MARCO, the BLEU-4 score is normalized across all questions, essentially giving different weights to different questions, while the ROUGE-L is averaged across different questions. We mainly focus on ROUGE-L as each question in a reading comprehension

Model	ROUGE-L	BLEU-4
BiDAF baseline	37.68	35.51
+ passage extraction	44.57	38.03
+ rich feature	**48.93**	**42.17**

Table 1: The influence of passage extraction and rich-feature representation on the development set

Loss	ROUGE-L	Δ
single answer	48.93	-
L_{min}	49.05	+ 0.12
L_{avg}	49.67	+ 0.74
L_{wavg}	**49.77**	**+ 0.84**

Table 2: Comparison among different choices for the loss function with multiple answers on the development set

dataset should have equal weight in evaluation (Tan et al., 2018). For a single question with multiple reference answers, the maximum score with any reference answers is used, as implemented in the official tool for ROUGE-L. This is reasonable as providing one valid answer is good enough in many cases.

4.2 Implementation Details

4.2.1 Word and POS Tag Embedding

We train a segmentation model with one-layer BiLSTM using the DuReader dataset, and apply it to a subset of SogouT corpus[2], which contains a large amount of Chinese web pages(Liu et al., 2012). 256-dimension word embeddings are trained on this data with language model task using one-layer BiLSTM model.

As for POS tag, we use a POS tagger trained on the Chinese Treebank (CTB) data to tag each word in questions and passages in the DuReader dataset. 64-dimension POS tag embeddings are trained on this data using one-layer BiLSTM model.

We keep all word and POS tag embeddings fixed during training.

4.2.2 Training and Parameters

The maximum length of each passage is set to be 500. The batch size is set to be 32. The dimension of hidden vector is set to be 150 for all layers. Dropout (Srivastava et al., 2014) is applied between layers, with a dropout rate of 0.15. We use $\lambda_{ps} = 3$ for passage selection loss and $\lambda = 10$ for MRT. All parameters are tuned on the DuReader development set.

[2] http://www.sogou.com/labs/resource/t.php

Model	ROUGE-L	Δ
single-answer baseline	48.93	-
+ multi-answer loss	49.77	+ 0.84
+ passage selection loss	49.96	+ 1.03
+ MRT	**50.62**	**+ 1.69**

Table 3: Results with multi-answer multi-task loss and Minimum Risk Training on the development set

As MRT training is more time-consuming than MLE training, our MRT model is initialized with model trained with MLE. It usually obtains the best result in just one epoch, which results in feasible training time.

Our model is optimized with Adam algorithm (Kingma and Ba, 2014), and the learning rate is fixed to 0.001 during training.

4.3 Results

4.3.1 Single-Answer Baseline

Table 1 shows the results for passage extraction and rich-feature representation (pre-trained word, POS, and word-in-query embeddings) on the development set. Both of them dramatically increase ROUGE-L and BLEU-4 score over the BiDAF baseline from the original DuReader paper. Together they form our single-answer baseline, on which we test the effectiveness of the multi-answer multi-task loss and Minimum Risk Training.

4.3.2 Different loss functions with multi-answer

Table 2 shows the experimental results with three different multi-answer loss functions introduced in Section 3.5.1. All of them offer improvement over the single-answer baseline, which shows the effectiveness of utilizing multiple answers. The average loss performs better than the min loss, which suggests that forcing the model to predict all possible answers is better than asking it to just find the easiest one. Not surprisingly, by taking into account the quality of different answer spans, the weighted average loss outperforms the average loss and achieves the best result among the three. All later experiments are conducted based on the weighted average loss.

Model	ROUGE-L	BLEU-4
BiDAF (He et al., 2017)	39.0	31.8
Match-LSTM (He et al., 2017)	39.2	31.9
PR+BiDAF (Wang et al., 2018b)	41.81	37.55
PE+BiDAF (ours)	45.93	38.86
V-Net (Wang et al., 2018b)	44.18	40.97
Our complete model	**51.09**	**43.76**
Human	57.4	56.1

Table 4: Performance of our model and competing models on the DuReader test set

Model	ROUGE-L	
	Q_s	Q_m
single-answer	38.01	53.8
multi-answer	**38.66**	**54.65**

Table 5: Results on Q_s and Q_m

4.3.3 Multi-task Loss and Minimum Risk Training

As we can see in Table 3, the ROUGE-L score on the DuReader development set increases to 49.77 by incorporating multi-answer into the loss function. Joint learning with passage selection loss yields an increase of 0.19. And with Minimum Risk Training, our model can reach a ROUGE-L score of 50.62, with a further increment of 0.66.

4.3.4 Comparison with State-of-the-art

Table 4 shows the performance of our model and other state-of-the-art models on the DuReader test set. First, we compare our passage extraction method with the paragraph ranking model from Wang et al. (2018b). Based on the same BiDAF model described in Section 3.4, our method (PE+BiDAF) significantly outperforms the trained model from Wang et al. (2018b) (PR+BiDAF) on the DuReader test set. As we can see, our complete model achieves the state-of-the-art performance in both ROUGE-L and BLEU-4, and greatly narrows the performance gap between MRC system and human in the challenging real-world setting.

4.3.5 Further Analysis

For further analysis, we construct two sets from the development set. Q_s contains 2787 questions with a single reference answer, and Q_m contains 6650 questions with more than one reference answer. 563 questions from the development set are labeled with no answer, and thus not included in Q_s or Q_m. Table 5 shows the performance of our model on Q_s and Q_m. It can be seen that even questions with single answer (Q_s) can benefit from using multiple answers in training. The improvement for Q_m is higher than that for Q_s.

5 Conclusion

In this paper, we focus on real-world machine reading comprehension. We propose a multi-answer multi-task framework to tackle the multi-answer problem which is common in everyday world. Minimum Risk Training is applied to solve the multi-occurrence problem of the answer. We also propose a simple method for passage extraction which solves the length issue of the passage. Experimental results indicate that our model achieves state-of-the-art performance in the challenging DuReader dataset.

Despite using multiple answers in training, our system only predicts a single answer in decoding time. However, in some cases (e.g. for questions about opinion), finding all possible answers may be desirable. In the future, we plan to design models which could generate all possible answers for a single question.

References

Ayana, Shiqi Shen, Zhiyuan Liu, and Maosong Sun. 2016. Neural headline generation with minimum risk training. *Computing Research Repository*, arXiv:1604.01904. Version 2.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1870–1879. Association for Computational Linguistics.

Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension.

Computing Research Repository, arXiv:1710.10723. Version 2.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2017. Attention-over-attention neural networks for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 593–602. Association for Computational Linguistics.

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

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

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

Minghao Hu, Yuxing Peng, and Xipeng Qiu. 2017. Reinforced mnemonic reader for machine comprehension. *Computing Research Repository*, arXiv:1705.02798. Version 5.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. Triviaqa: A large scale distantly supervised challenge dataset for reading comprehension. *Computing Research Repository*, arXiv:1705.03551.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *Computing Research Repository*, arXiv:1412.6980.

Yiqun Liu, Fei Chen, Weize Kong, Huijia Yu, Min Zhang, Shaoping Ma, and Liyun Ru. 2012. Identifying web spam with the wisdom of the crowds. *ACM Transactions on the Web (TWEB)*, 6(1):2.

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. *Computing Research Repository*, arXiv:1611.09268.

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*, pages 2383–2392. Association for Computational Linguistics.

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

Shiqi Shen, Yong Cheng, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Minimum risk training for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1683–1692. Association for Computational Linguistics.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. 2014. Dropout: A simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958.

Chuanqi Tan, Furu Wei, Nan Yang, Bowen Du, Weifeng Lv, and Ming Zhou. 2018. S-net: From answer extraction to answer synthesis for machine reading comprehension.

Shuohang Wang and Jing Jiang. 2017. Machine comprehension using match-lstm and answer pointer. In *Proceedings of International Conference on Learning Representations*.

Shuohang Wang, Mo Yu, Jing Jiang, Wei Zhang, Xiaoxiao Guo, Shiyu Chang, Zhiguo Wang, Tim Klinger, Gerald Tesauro, and Murray Campbell. 2018a. Evidence aggregation for answer re-ranking in open-domain question answering. In *Proceedings of International Conference on Learning Representations*.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. 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)*, pages 189–198. Association for Computational Linguistics.

Yizhong Wang, Kai Liu, Jing Liu, Wei He, Yajuan Lyu, Hua Wu, Sujian Li, and Haifeng Wang. 2018b. Multi-passage machine reading comprehension with cross-passage answer verification. *Computing Research Repository*, arXiv:1805.02220.

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)*, pages 271–280. Association for Computational Linguistics.

Caiming Xiong, Victor Zhong, and Richard Socher. 2017. Dynamic coattention networks for question answering. In *Proceedings of International Conference on Learning Representations*.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 2013–2018. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Minh-Thang Luong, Rui Zhao, Kai Chen, Mohammad Norouzi, and Quoc V Le. 2018. Qanet: Combining local convolution with global self-attention for reading comprehension. In *Proceedings of International Conference on Learning Representations*.