DCMN+: Dual Co-Matching Network for Multi-Choice Reading Comprehension

Shuailiang Zhang,[1,2,3] **Hai Zhao,**[1,2,3*] **Yuwei Wu,**[1,2,3]
Zhuosheng Zhang,[1,2,3] **Xi Zhou,**[4] **Xiang Zhou**[4]
[1]Department of Computer Science and Engineering, Shanghai Jiao Tong University
[2]Key Laboratory of Shanghai Education Commission for Intelligent Interaction
and Cognitive Engineering, Shanghai Jiao Tong University, Shanghai, China
[3]MoE Key Lab of Artificial Intelligence, AI Institute, Shanghai Jiao Tong University, Shanghai, China
[4]CloudWalk Technology, Shanghai, China
{zsl123, will8821, zhangzs}@sjtu.edu.cn, zhaohai@cs.sjtu.edu.cn,
{zhouxi, zhouxiang}@cloudwalk.cn

Abstract

Multi-choice reading comprehension is a challenging task to select an answer from a set of candidate options when given passage and question. Previous approaches usually only calculate question-aware passage representation and ignore passage-aware question representation when modeling the relationship between passage and question, which cannot effectively capture the relationship between passage and question. In this work, we propose dual co-matching network (DCMN) which models the relationship among passage, question and answer options bidirectionally. Besides, inspired by how humans solve multi-choice questions, we integrate two reading strategies into our model: (i) passage sentence selection that finds the most salient supporting sentences to answer the question, (ii) answer option interaction that encodes the comparison information between answer options. DCMN equipped with the two strategies (DCMN+) obtains state-of-the-art results on five multi-choice reading comprehension datasets from different domains: RACE, SemEval-2018 Task 11, ROCStories, COIN, MCTest.

Passage: *Runners in a relay race pass a stick in one direction. However, merchants passed silk, gold, fruit, and glass along the Silk Road in more than one direction. They earned their living by traveling the famous Silk Road. ...* ***The Silk Road was made up of many routes, not one smooth path.*** *They passed through what are now 18 countries. The routes crossed mountains and deserts and had many dangers of hot sun, deep snow and even battles...*

Question: *The Silk Road became less important because _ .*

A. *it was made up of different routes*
B. *silk trading became less popular*
C. ***sea travel provided easier routes***
D. *people needed fewer foreign goods*

Table 1: An example passage with related question and options from RACE dataset. The ground-truth answer and the evidence sentences in the passage are in **bold**.

Introduction

Machine reading comprehension (MRC) is a fundamental and long-standing goal of natural language understanding which aims to teach machine to answer question automatically according to given passage (Hermann et al. 2015; Rajpurkar et al. 2016; Nguyen et al. 2016; Zhang et al. 2018). In this paper, we focus on multi-choice MRC tasks such as RACE (Lai et al. 2017) which requests to choose the right option from a set of candidate answers according to given passage and question. Different from MRC datasets such as SQuAD (Rajpurkar et al. 2016) and NewsQA (Trischler et al. 2017) where the expected answer is usually in the form of a short span from the given passage, answer in multi-choice MRC is non-extractive and may not appear in the original passage, which allows rich types of questions such as commonsense reasoning and passage summarization, as illustrated by the example in Table 1.

Pre-trained language models such as BERT (Devlin et al. 2019) and XLNet (Yang et al. 2019) have achieved significant improvement on various MRC tasks. Recent works on MRC may be put into two categories, training more powerful language models or exploring effective applying pattern of the language models to solve specific task. There is no doubt that training a better language model is essential and indeed extremely helpful (Devlin et al. 2019; Yang et al. 2019) but at the same time it is time-consuming and resource-demanding to impart massive amounts of general knowledge from external corpora into a deep language model via pre-training (Sun et al. 2019; Zhang et al. 2019b). For example, training a 24-layer transformer (Devlin et al. 2019) requires 64 TPUs for 4 days. So from the practical viewpoint, given limited computing resources and a well pre-trained model, can we improve the machine reading comprehension during fine-tuning instead of via expensive full pre-training? This work starts from this viewpoint and focuses on exploring effective applying pattern of lan-

[*] Corresponding author. This paper was partially supported by National Key Research and Development Program of China (No. 2017YFB0304100), Key Projects of National Natural Science Foundation of China (U1836222 and 61733011).

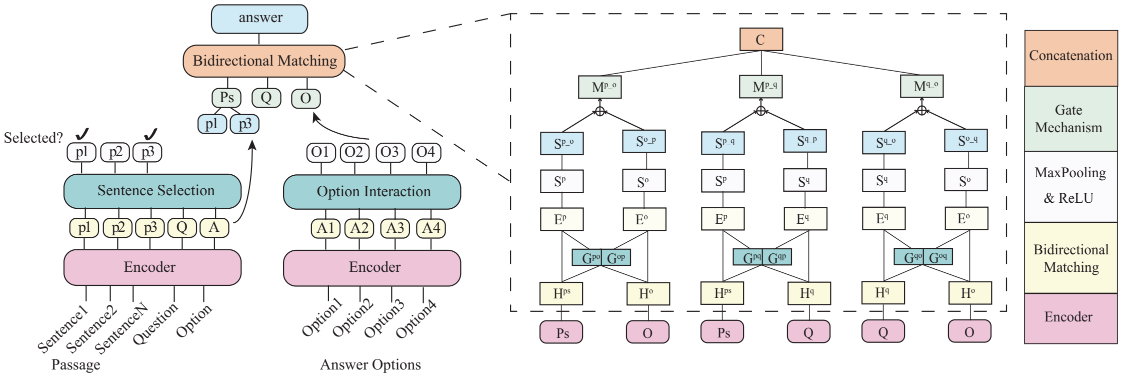

Figure 1: The framework of our model. P-Passage, Q-Question, O-Option.

guage models instead of presenting better language models to furthermore enhance state-of-the-art multi-choice MRC. We will show the way to use a strong pre-trained language model may still have a heavy impact on MRC performance no matter how strong the language model itself is.

To well handle multi-choice MRC problem, an effective solution has to carefully model the relationship among the triplet of three sequences, passage (**P**), question (**Q**) and answer candidate options (**A**) with a matching module to determine the answer. However, previous unidirectional matching strategies usually calculate question-aware passage representation and ignore passage-aware question representation when modeling the relationship between passage and question (Wang et al. 2018b; Tang, Cai, and Zhuo 2019; Chen et al. 2019).

To alleviate such an obvious defect in modeling the $\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}$ triplet from existing work, we propose dual co-matching network (DCMN) which bidirectionally incorporates all the pairwise relationships among the $\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}$ triplet. In detail, we model the passage-question, passage-option and question-option pairwise relationship simultaneously and bidirectionally for each triplet, exploiting the gated mechanism to fuse the representations from two directions. Besides, we integrate two reading strategies which humans usually use into the model. One is passage sentence selection that helps extract salient evidence sentences from the given passage, and then matches evidence sentences with answer options. The other is answer option interaction that encodes comparison information into each option. The overall framework is shown in Figure 1. The output of pre-trained language model (i.e. BERT (Devlin et al. 2019) and XLNet (Yang et al. 2019)) is used as the contextualized encoding. After passage sentence selection and answer option interaction, bidirectional matching representations are built for every pairwise relationship among the $\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}$ triplet.

Our model achieves new state-of-the-art results on the multi-choice MRC benchmark challenge RACE (Lai et al. 2017). We further conduct experiments on four representative multi-choice MRC datasets from different domains (i.e., ROCStories (Mostafazadeh et al. 2016), SemEval-2018 Task 11 (Ostermann et al. 2018), MCTest (Richardson, Burges, and Renshaw 2013), COIN Shared Task 1 (Ostermann et al. 2018)) and achieve the absolute improvement of 4.9% and 2.8% in average accuracy from directly fine-tuned BERT and XLNet, respectively, which indicates our method has a heavy impact on the MRC performance no matter how strong the pre-trained language model itself is.

Our Proposed Model

The illustration of our model is shown in Figure 1. The major components of the model are Contextualized Encoding, Passage Sentence Selection, Answer Option Interaction and Bidirectional Matching. We will discuss each component in detail.

Task Definition

For the task of multi-choice reading comprehension, the machine is given a passage (**P**), a question (**Q**), and a set of answer candidate options (**A**) to select the correct answer from the candidates, where $\mathbf{P} = \{\mathbf{p_1}, \mathbf{p_2}, ..., \mathbf{p_n}\}$ is the passage composed of n sentences, $\mathbf{A} = \{\mathbf{A_1}, \mathbf{A_2}, ..., \mathbf{A_m}\}$ is the option set with m answer candidates.

Contextualized Encoding

In this work, pre-trained language models are used as the encoder of our model which encodes each token in passage and question into a fixed-length vector. Given an encoder, the passage, the question, and the answer options are encoded as follows:

$$\begin{aligned}\mathbf{H}^p = Encode(\mathbf{P}), \mathbf{H}^q = Encode(\mathbf{Q})\\ \mathbf{H}^a = Encode(\mathbf{A})\end{aligned} \tag{1}$$

where $Encode(\cdot)$ returns the last layer output by the encoder, which can be well pre-trained language models such as BERT (Devlin et al. 2019) and XLNet (Yang et al. 2019), as using transformer as the contextualized encoder has shown to be very powerful in language representation (Zhang et al. 2019a; Zhou and Zhao ; Luo, Xiao, and Zhao 2019; Xiao et al. 2019). $\mathbf{H}^p \in R^{|P| \times l}$, $\mathbf{H}^q \in R^{|Q| \times l}$, and $\mathbf{H}^a \in R^{|A| \times l}$ are sequence representation of passage, question and answer option, respectively. $|P|$, $|Q|$, $|A|$ are the sequence length, respectively. l is the dimension of the hidden state.

N sent	% on RACE	% on COIN	Passage	Question
1	65	76	Soon after, the snack came out. *I then opened the chips and started to enjoy them, before enjoying the* ***soda***. I had a great little snack...	*What else did the person enjoy?*
2	22	10	*She lived in the house across the street that had 9 people and* ***3 dogs and another cat*** *living in it. She didn't seem very happy there, especially with a 2 year old that chased her and grabbed her.* The other people in the house agreed...	*What did the 2 year old's mom own?*
>=3	13	14	*When I was hungry last week for a little snack and a soda, I went to the closest vending machine. I felt that it was a little overpriced, but being as though I needed something...*	*What's the main idea of this passage?*

Table 2: Analysis of the sentences in passage required to answer questions on RACE and COIN. 50 examples from each dataset are sampled randomly. N sent indicates the number of sentences required to answer the question. The evidence sentences in the passage are in *emphasis* and the correct answer is with **bold**.

Passage Sentence Selection

Existing multi-choice MRC models learn the passage representation with all the sentences in one-shot, which is inefficient and counter-intuitive. To explore how many sentences are necessarily required to answer the question, we randomly extract 50 examples from the development set of RACE and COIN, as shown in Table 2. Among all examples, 87% questions on RACE and 86% on COIN can be answered within two evidence sentences. From this observation, the model should be extremely beneficial if focusing on a few key evidence sentences.

To select the evidence sentences from the passage $\mathbf{P} = \{\mathbf{p_1}, \mathbf{p_2}, .., \mathbf{p_i}, .., \mathbf{p_n}\}$, this module scores each sentence p_i with respect to the question $\mathbf{Q}$ and answer option $\mathbf{A}$ in parallel. The top K scored sentences will be selected. This module shares the encoder with the whole model. For each $\{\mathbf{p_i}, \mathbf{Q}, \mathbf{A}\}$ triplet, $\mathbf{H}^{p_i} \in R^{|p_i|\times l}$, $\mathbf{H}^q$, and $\mathbf{H}^a$ are all representations offered by the encoder. Here we introduce two methods to compute the score of the triplet based on the representations.

- **Cosine score**: The model computes word-by-word cosine similarity between the sentence and question-option sequence pair.

$$\begin{aligned}
\mathbf{D}^{pa} &= Cosine(\mathbf{H}^a, \mathbf{H}^{p_i}) \in R^{|A|\times|p_i|} \\
\mathbf{D}^{pq} &= Cosine(\mathbf{H}^q, \mathbf{H}^{p_i}) \in R^{|Q|\times|p_i|} \\
\bar{\mathbf{D}}^{pa} &= MaxPooling(\mathbf{D}^{pa}) \in R^{|A|} \\
\bar{\mathbf{D}}^{pq} &= MaxPooling(\mathbf{D}^{pq}) \in R^{|Q|} \\
score &= \frac{\sum_{k=1}^{|A|} \bar{\mathbf{D}}_k^{pa}}{|A|} + \frac{\sum_{k=1}^{|Q|} \bar{\mathbf{D}}_k^{pq}}{|Q|}
\end{aligned} \quad (2)$$

where $\mathbf{D}^{pa}$, $\mathbf{D}^{pa}$ are the distance matrices and $\mathbf{D}_{ij}^{pa}$ is the cosine similarity between the i-th word in the candidate option and the j-th word in the passage sentence.

- **Bilinear score**: Inspired by (Min et al. 2018a), we compute the bilinear weighted distance between two sequences, which can be calculated as follows:

$$\begin{aligned}
\alpha &= SoftMax(\mathbf{H}^q W_1) \in R^{|Q|\times l} \\
\mathbf{q} &= \alpha^T \mathbf{H}^q \in R^l \\
\bar{\mathbf{P}}_j &= \mathbf{H}_j^{p_i} W_2 \mathbf{q} \in R^l, j \in [1, |p_i|] \\
\hat{\mathbf{P}}^{pq} &= Max(\bar{\mathbf{P}}_1 \bar{\mathbf{P}}_2, ..., \bar{\mathbf{P}}_{|p_i|}) \in R^l
\end{aligned} \quad (3)$$

where W_1, $W_2 \in R^{l\times l}$ are learnable parameters, $\hat{\mathbf{P}}^{pq}$ is the bilinear similarity vector between the passage sentence and question. Similarly, the vector $\hat{\mathbf{P}}^{pa}$ between the passage sentence and answer can be calculated with the same procedure. The final score can be computed as follows:

$$score = W_3^T \hat{\mathbf{P}}^{pq} + W_4^T \hat{\mathbf{P}}^{pa} \quad (4)$$

where W_3, $W_4 \in R^l$ are learnable parameters.

After scoring each sentence, top K scored sentences are selected and concatenated together as an updated passage $\mathbf{P}_s$ to replace original full passage. So the new sequence triplet is $\{\mathbf{P}_s, \mathbf{Q}, \mathbf{A}\}$ and the new passage is represented as $\mathbf{H}^{ps}$.

Answer Option Interaction

Human solving multi-choice problem may seek help from comparing all answer options. For example, one option has to be picked up not because it is the most likely correct, but all the others are impossibly correct. Inspired by such human experience, we introduce the comparison information among answer options so that each option is not independent of the other. Here we build bilinear representations between any two options. Gated mechanism (Srivastava, Greff, and Schmidhuber 2015) is used to fuse interaction representation into the original answer option representations.

The encoder encodes each answer option $\mathbf{A}_i$ as $\mathbf{H}^{a_i}$. Then the comparison vector between option $\mathbf{A}_i$ and $\mathbf{A}_j$ can be computed as follows:

$$\begin{aligned}
\mathbf{G} &= SoftMax(\mathbf{H}^{a_i} W_5 \mathbf{H}^{a_j T}) \in R^{|A_i|\times|A_j|} \\
\mathbf{H}^{a_{i,j}} &= ReLU(\mathbf{G}\mathbf{H}^{a_j}) \in R^{|A_i|\times l}
\end{aligned} \quad (5)$$

where $W_5 \in R^{l\times l}$ is one learnable parameter, $\mathbf{G}$ is the bilinear interaction matrix between A_i and A_j, $\mathbf{H}^{a_{i,j}}$ is the interaction representation. Then gated mechanism is used to fuse interaction representation into the original answer option representations as follows:

$$
\begin{aligned}
\hat{\mathbf{H}}^{a_i} &= [\{\mathbf{H}^{a_{i,j}}\}_{j\neq i}] \in R^{|A_i|\times(m-1)l} \\
\bar{\mathbf{H}}^{a_i} &= \hat{\mathbf{H}}^{a_i} W_6 \in R^{|A_i|\times l} \\
g &= \sigma(\bar{\mathbf{H}}^{a_i} W_7 + \mathbf{H}^{a_i} W_8 + b) \\
\mathbf{H}^{o_i} &= g * \mathbf{H}^{a_i} + (1-g) * \bar{\mathbf{H}}^{a_i}
\end{aligned}
\tag{6}
$$

where W_7, $W_8 \in R^{l\times l}$ and $W_6 \in R^{(m-1)l\times l}$ are learnable parameters, $\hat{\mathbf{H}}^{a_i}$ is the concatenation of all the interaction representations. $g \in R^{|A_i|\times l}$ is a reset gate which balances the influence of $\bar{\mathbf{H}}^{a_i}$ and $\mathbf{H}^{a_i}$, and $\mathbf{H}^{o_i}$ is the final option representation of $\mathbf{A}_i$ encoded with the interaction information. At last, we denote $\mathbf{O} = \{\mathbf{H}^{o_1}, \mathbf{H}^{o_2}, ..., \mathbf{H}^{o_m}\}$ as the final answer option representation set fused with comparison information across answer options.

Bidirectional Matching

The triplet changes from $\{\mathbf{P}, \mathbf{Q}, \mathbf{A}\}$ to $\{\mathbf{P}_s, \mathbf{Q}, \mathbf{O}\}$ with passage sentence selection and answer option interaction. To fully model the relationship in the $\{\mathbf{P}_s, \mathbf{Q}, \mathbf{O}\}$ triplet, bidirectional matching is built to get all pairwise representations among the triplet, including passage-answer, passage-question and question-answer representation. Here shows how to model the relationship between question-answer sequence pair as an example and it is the same for the other two pairs.

Bidirectional matching representation between the question $\mathbf{H}^q$ and answer option $\mathbf{H}^o$ can be calculated as follows:

$$
\begin{aligned}
\mathbf{G}^{qo} &= SoftMax(\mathbf{H}^q W_9 \mathbf{H}^{o^T}) \\
\mathbf{G}^{oq} &= SoftMax(\mathbf{H}^o W_{10} \mathbf{H}^{q^T}) \\
\mathbf{E}^q &= \mathbf{G}^{qo}\mathbf{H}^o, \mathbf{E}^o = \mathbf{G}^{oq}\mathbf{H}^q \\
\mathbf{S}^q &= ReLU(\mathbf{E}^q W_{11}) \\
\mathbf{S}^o &= ReLU(\mathbf{E}^o W_{12})
\end{aligned}
\tag{7}
$$

where W_9, W_{10}, W_{11}, $W_{12} \in R^{l\times l}$ are learnable parameters. $\mathbf{G}^{qo} \in R^{|Q|\times|O|}$ and $\mathbf{G}^{oq} \in R^{|O|\times|Q|}$ are the weight matrices between question and answer option. $\mathbf{E}^q \in R^{|Q|\times l}, \mathbf{E}^o \in R^{|A|\times l}$ represent option-aware question representation and question-aware option representation, respectively. The final representation of question-answer pair is calculated as follows:

$$
\begin{aligned}
\mathbf{S}^{q\text{-}o} &= MaxPooling(\mathbf{S}^q) \\
\mathbf{S}^{o\text{-}q} &= MaxPooling(\mathbf{S}^o) \\
g &= \sigma(\mathbf{S}^{q\text{-}o} W_{13} + \mathbf{S}^{o\text{-}q} W_{14} + b) \\
\mathbf{M}^{q\text{-}o} &= g * \mathbf{S}^{o\text{-}q} + (1-g) * \mathbf{S}^{o\text{-}q}
\end{aligned}
\tag{8}
$$

where $W_{13}, W_{14} \in R^{l\times l}$ and $b \in R^l$ are three learnable parameters. After a row-wise max pooling operation, we get the aggregation representation $\mathbf{M}^q \in R^l$ and $\mathbf{M}^o \in R^l$. $g \in R^l$ is a reset gate. $\mathbf{M}^{q\text{-}o} \in R^l$ is the final bidirectional matching representation of the question-answer sequence pair.

Passage-question and passage-option sequence matching representation $\mathbf{M}^{p\text{-}q}, \mathbf{M}^{p\text{-}o} \in R^l$ can be calculated in the same procedure from Eq.(7) to Eq.(8). The framework of this module is shown in Figure 1.

Objective Function

With the built matching representations $\mathbf{M}^{p\text{-}q}, \mathbf{M}^{p\text{-}o}, \mathbf{M}^{q\text{-}o}$ for three sequence pairs, we concatenate them as the final representation $\mathbf{C} \in R^{3l}$ for each passage-question-option triplet. We denote the representation $\mathbf{C}_i$ for each $\{P_s, Q, O_i\}$ triplet. If A_k is the correct option, then the objective function can be computed as follows:

$$
\begin{aligned}
\mathbf{C} &= [\mathbf{M}^{p\text{-}q}; \mathbf{M}^{p\text{-}o}; \mathbf{M}^{q\text{-}o}] \\
L(A_k|P,Q) &= -log\frac{\exp(V^T\mathbf{C}_k)}{\sum_{j=1}^{m}\exp(V^T\mathbf{C}_j)}
\end{aligned}
\tag{9}
$$

where $V \in R^{3l}$ is a learnable parameter and m is the number of answer options.

Experiments

Dataset

We evaluate our model on five multi-choice MRC datasets from different domains. Statistics of these datasets are detailed in Table 3. Accuracy is calculated as $acc = N^+/N$, where N^+ and N are the number of correct predictions and the total number of questions. Some details about these datasets are shown as follows:

- **RACE** (Lai et al. 2017): RACE consists of two subsets: RACE-M and RACE-H respectively corresponding to middle school and high school difficulty levels, which is recognized as one of the largest and most difficult datasets in multi-choice reading comprehension.
- **SemEval-2018 Task11** (Ostermann et al. 2018): Multi-choice questions should be answered based on narrative texts about everyday activities.
- **ROCStories** (Mostafazadeh et al. 2016): This dataset contains 98,162 five-sentence coherent stories in the training dataset, 1,871 four-sentence story contexts along with a right ending and a wrong ending in the development and test datasets, respectively.
- **MCTest** (Richardson, Burges, and Renshaw 2013): This task requires machines to answer questions about fictional stories, directly tackling the high-level goal of open-domain machine comprehension.

Task	Domain	#o	#p	#q
RACE	general	4	27,933	**97,687**
SemEval	narrative text	2	2,119	13,939
ROCStories	stories	2	3472	3472
MCTest	stories	4	660	2,640
COIN	everyday scenarios	2	_	5,102

Table 3: Statistics of multi-choice machine reading comprehension datasets. #o is the average number of candidate options for each question. #p and #q are the number of documents and questions in the dataset.

Model	RACE-M/H	RACE
HAF (Zhu et al. 2018)	45.0/46.4	46.0
MRU (Tay, Tuan, and Hui 2018)	57.7/47.4	50.4
HCM (Wang et al. 2018b)	55.8/48.2	50.4
MMN (Tang, Cai, and Zhuo 2019)	61.1/52.2	54.7
GPT (Radford 2018)	62.9/57.4	59.0
RSM (Sun et al. 2019)	69.2/61.5	63.8
OCN (Ran et al. 2019)	76.7/69.6	71.7
XLNet (Yang et al. 2019)	85.5/80.2	81.8
BERT$_{\text{base}}$*	71.1/62.3	65.0
BERT$_{\text{large}}$*	76.6/70.1	72.0
XLNet$_{\text{large}}$*	83.7/78.6	80.1
Our Models		
BERT$_{\text{base}}$* + DCMN	73.2/64.2	67.0
BERT$_{\text{large}}$* + DCMN	79.2/72.1	74.1
BERT$_{\text{large}}$* + DCMN + P$_{\text{SS}}$ + A$_{\text{OI}}$	79.3/74.4	**75.8**
XLNet$_{\text{large}}$* + DCMN + P$_{\text{SS}}$ + A$_{\text{OI}}$	86.5/81.3	**82.8**
Human Performance		
Turkers	85.1/69.4	73.3
Ceiling	95.4/94.2	94.5

Table 4: Experiment results on RACE test set. All the results are from single models. P$_{\text{SS}}$: Passage Sentence Selection; A$_{\text{OI}}$: Answer Option Interaction. * indicates our implementation.

- **COIN Task 1** (Ostermann et al. 2018): The data for the task is short narrations about everyday scenarios with multiple-choice questions.

Implementation Details

Our model is evaluated based on the pre-trained language model BERT (Devlin et al. 2019) and XLNet (Yang et al. 2019) which both have small and large versions. The basic version BERT$_{\text{base}}$ has 12-layer transformer blocks, 768 hidden-size, and 12 self-attention heads, totally 110M parameters. The large version BERT$_{\text{large}}$ has 24-layer transformer blocks, 1024 hidden-size, and 16 self-attention heads, totally 340M parameters. Two versions of XLNet have the similar sizes as BERT.

In our experiments, the max input sequence length is set to 512. A dropout rate of 0.1 is applied to every BERT layer. We optimize the model using BertAdam (Devlin et al. 2019) optimizer with a learning rate 2e-5. We train for 10 epochs with batch size 8 using eight 1080Ti GPUs when BERT$_{\text{large}}$ and XLNet$_{\text{large}}$ are used as the encoder. Batch size is set to 16 when using BERT$_{\text{base}}$ and XLNet$_{\text{base}}$ as the encoder[1].

Evaluation and Ablation Study on RACE

Table 4 reports the experimental results on RACE and its two subtasks: RACE-M and RACE-H. In the table, Turkers is the performance of Amazon Turkers on a randomly sampled subset of the RACE test set and Ceiling is the percentage of the unambiguous questions with a correct answer in a subset of the test set. Here we give the results of directly fine-tuned BERT$_{\text{base}}$, BERT$_{\text{large}}$ and XLNet$_{\text{large}}$ on RACE and get the accuracy of 65.0%, 72.0% and 80.1%, respectively. Because of the limited computing resources, the largest batch size can only be set to 8 in our experiments which leads to 1.7% decrease (80.1% vs. 81.8%) on XLNet compared to the result reported in (Yang et al. 2019)[2].

	BERT$_{\text{base}}$	BERT$_{\text{large}}$	XLNet$_{\text{large}}$
base encoder	64.6	71.8	80.1
+ DCMN	66.0 (+1.4)	73.8 (+2.0)	81.5 (+1.4)
+ DCMN + P_SS	66.6 (+2.0)	74.6 (+2.8)	82.1 (+2.0)
+ DCMN + P_OI	66.8 (+2.2)	74.4 (+2.6)	82.2 (+2.1)
+ DCMN + ALL (DCMN+)	**67.4 (+2.8)**	**75.4 (+3.6)**	**82.6 (+2.5)**

Table 5: Ablation study on RACE dev set. P$_{\text{SS}}$: Passage Sentence Selection. A$_{\text{OI}}$: Answer Option Interaction. DCMN+: DCMN + P$_{\text{SS}}$ + A$_{\text{OI}}$.

The comparison indicates that our proposed method obtains significant improvement over pre-trained language models (75.8% vs. 72.0% on BERT$_{\text{large}}$ and 82.8% vs. 80.1% on XLNet$_{\text{large}}$) and achieves the state-of-the-art result on RACE.

In Table 5, we focus on the contribution of main components (DCMN, passage sentence selection and answer option interaction) in our model. From the results, the bidirectional matching strategy (DCMN) gives the main contribution and achieves further improvement by integrating with the two reading strategies. Finally, we have the best performance by combining all components.

Evaluation on Other Multi-choice Datasets

The results on four other multi-choice MRC challenges are shown in Table 6. When adapting our method to the non-conventional MRC dataset ROCStories which requires to choose the correct ending to a four-sentence incomplete story from two answer options (Mostafazadeh et al. 2016), the question context is left empty as no explicit questions are provided. Passage sentence selection is not used in this dataset because there are only four sentences as the passage. Since the test set of COIN is not publicly available, we report the performance of the model on its development set.

As shown in Table 6, we achieve state-of-the-art (SOTA) results on all datasets and obtain 3.1% absolute improvement in average accuracy over the previous average SOTA (88.9% vs. 85.8%) by using BERT as encoder and 4.8% (90.6% vs. 85.8%) by using XLNet as encoder. To further investigate the contribution of our model, we also report the results of directly fine-tuned BERT/XLNet on the target datasets. From the comparison, we can see that our model obtains 4.9% and 2.8% absolute improvement in average accuracy over the baseline of directly fine-tuned BERT (88.9% vs. 84.0%) and XLNet (90.6% vs. 87.8%), respectively. These results indicate our proposed model has a heavy impact on the performance no matter how strong the adopted pre-trained language model itself is.

Comparison with Unidirectional Methods

Here we focus on whether the bidirectional matching method works better than previous unidirectional methods.

[1] Our code is at https://github.com/Qzsl123/dcmn.

[2] The implementation is very close to the result 80.3% in (Yang et al. 2019) when using batch size 8 on RACE.

Task	Previous STOA		BERT	DCMN_BERT	XLNet	DCMN_XLNet
SemEval Task 11	(Sun et al. 2019)	89.5	90.5	91.8 (+1.3)	92.0	93.4 (+1.4)
ROCStories	(Li, Ding, and Liu 2019)	91.8	90.8	92.4 (+1.6)	93.8	95.8 (+2.0)
MCTest-MC160	(Sun et al. 2019)	81.7	73.8	85.0 (+11.2)	80.6	86.2 (+5.6)
MCTest-MC500	(Sun et al. 2019)	82.0	80.4	86.5 (+6.1)	83.4	86.6 (+3.2)
COIN Task 1	(Devlin et al. 2019)	84.2	84.3	88.8 (+4.5)	89.1	91.1 (+2.0)
Average		85.8	84.0	**88.9 (+4.9)**	87.8	**90.6 (+2.8)**

Table 6: Results on the test set of SemEval Task 11, ROCStories, MCTest and the development set of COIN Task 1. The test set of COIN is not public. DCMN_BERT: BERT + DCMN + P_{SS} + A_{OI}. Previous SOTA: previous state-of-the-art model. All the results are from single models.

Model	RACE	Model	RACE	Model	RACE
$\text{BERT}_{\text{base}}$	64.6				
+ Unidirectional					
$[S^{P\text{-}O}; S^{P\text{-}Q}; S^{O\text{-}Q}]$	65.0	$[S^{P\text{-}Q}; S^{Q\text{-}O}]$	63.4	$[S^{P\text{-}Q}; S^{O\text{-}Q}]$	64.5
$[S^{P\text{-}O}; S^{Q\text{-}P}; S^{O\text{-}Q}]$	65.2	$[S^{P\text{-}O}; S^{Q\text{-}O}]$	63.6	$[S^{Q\text{-}P}; S^{O\text{-}Q}]$	65.2
$[S^{P\text{-}Q}; S^{P\text{-}O}]$ (HCM)	64.4	$[S^{P\text{-}O}; S^{O\text{-}Q}]$	64.2	$[S^{P\text{-}Q}; S^{O\text{-}P}]$	64.7
$[S^{P\text{-}O}; S^{P\text{-}Q}; S^{Q\text{-}O}]$ (HAF)	64.2	$[S^{P\text{-}O}; S^{Q\text{-}P}; S^{Q\text{-}O}]$	64.4	$[S^{Q\text{-}P}; S^{Q\text{-}O}]$	64.3
$[S^{Q\text{-}O}; S^{O\text{-}Q}; S^{P\text{-}Q}; S^{P\text{-}O}]$ (MMN)	63.2				
+ Bidirectional					
$[M^{P\text{-}Q}; M^{P\text{-}O}]$	66.4	$[M^{P\text{-}O}; M^{Q\text{-}O}]$	66.0	$[M^{P\text{-}Q}; M^{Q\text{-}O}]$	65.5
$[M^{P\text{-}Q}; M^{P\text{-}O}; M^{Q\text{-}O}]$ (DCMN)	**67.1**				

Table 7: Performance comparison with different combination methods on the RACE dev set. (HCM) (Wang et al. 2018b), (HAF) (Zhu et al. 2018), (MMN) (Tang, Cai, and Zhuo 2019) are previous methods. We use $\text{BERT}_{\text{base}}$ as our encoder here. $[;]$ indicates the concatenation operation. $S^{P\text{-}O}$ and $M^{P\text{-}O}$ are the unidirectional and bidirectional representation referred in Eq. 8.

Top K	1	2	3	4	5	6
RACE-cos	58.4	60.1	63.3	65.8	**66.5**	66
RACE-bi	59.5	60.5	63.4	**66.8**	66.4	66.2
COIN-cos	81.0	82.0	**83.5**	83.0	82.5	82.4
COIN-bi	81.7	82.0	82.6	**82.8**	82.4	82.2

Table 8: Results on RACE and COIN dev set with cosine and bilinear score in P_{SS}. We use $\text{BERT}_{\text{base}}$ as encoder here.

In Table 7, we enumerate all the combinations of unidirectional matching strategies[3] which only use passage-aware question representation $S^{Q\text{-}P}$ or question-aware passage representation $S^{P\text{-}Q}$ when modeling the relationship between the passage and question. Specially, we roughly summarize the matching methods in previous work (i.e. HCM, HAF, MMN) using our model notations which meet their general ideas except some calculation details.

From the comparison, we observe that previous matching strategies (HCM 64.4%, HAF 64.2%, MMN 63.2%) fail to give further performance improvement over the strong encoder (64.6%). In contrast, all bidirectional combinations work better than the encoder. All three pairwise matching representations ($M^{P\text{-}Q}$, $M^{P\text{-}O}$, $M^{Q\text{-}O}$) are necessary and by concatenating them together, we achieve the highest performance (67.1%).

Results with Different Settings in P_{SS}

Table 8 shows the performance comparison with different scoring methods, and we observe that both methods have their advantages and disadvantages. Cosine score method works better on COIN dataset (83.5% vs. 82.8%) and bilinear score works better on RACE dataset (66.8% vs. 66.5%).

Figure 2 shows the results of passage sentence selection (P_{ss}) on COIN and RACE dev set with different numbers of selected sentences (Top K). The results without P_{ss} module are also shown in the figure (RACE-w and COIN-w). We observe that P_{ss} mechanism consistently shows a positive impact on both datasets when more than four sentences are selected compared to the model without P_{ss} (RACE-w and COIN-w). The highest performance is achieved when top 3 sentences are selected on COIN and top 5 sentences on RACE where the main reason is that the questions in RACE are designed by human experts and require more complex reasoning.

Why Previous Methods Break Down?

As shown in Table 7, applying previous models to a strong BERT encoder fails to give performance increase over directly fine-tuned BERT. The contrast is clear that our proposed model achieves more than 3.8% absolute increase over the BERT baseline. We summarize the reasons resulting in such contrast as follows: (i) the unidirectional representations cannot well capture the relationship between two sequences, (ii) previous methods use elementwise subtraction and multiplication to fuse $\mathbf{E}^q$ and $\mathbf{H}^o$ in Eq. 7 (i.e.,

[3] Here we omit the combinations with $S^{O\text{-}P}$ because we find the combinations with $S^{P\text{-}O}$ works better than $S^{O\text{-}P}$.

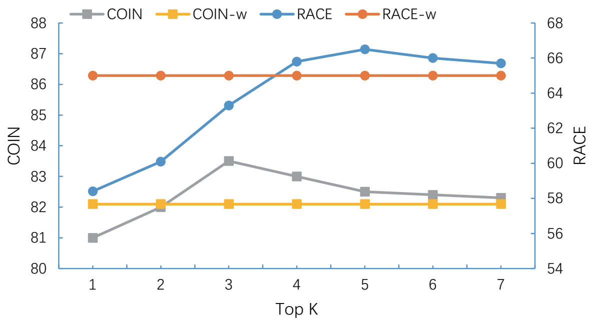

Figure 2: Results of sentence selection on dev sets of RACE and COIN when selecting different numbers of sentences (Top K). We use BERT$_{\text{base}}$ as encoder and cosine score method here. RACE/COIN-w indicates the results on RACE/COIN without passage sentence selection module.

$[\mathbf{E}^q \ominus \mathbf{H}^o; \mathbf{E}^q \otimes \mathbf{H}^o])$ which is shown suboptimal as such processing breaks the symmetry of equation. Symmetric representations from both directions show essentially helpful for bidirectional architecture.

Evaluation on Different Types of Questions

Inspired by (Sun et al. 2019), we further analyze the performance of the main components on different question types. Questions are roughly divided into five categories: *detail*, *inference*, *main*, *attitude* and *vocabulary* (Lai et al. 2017; Qian and Schedl 2004). We annotate all the instances of the RACE development set. As shown in Figure 3, all the combinations of components work better than the BERT baseline in most question types. Bidirectional matching strategy (DCMN) consistently improves the results across all categories. DCMN+P$_{\text{SS}}$ works best on the *inference* and *attitude* categories which indicates P$_{SS}$ module may effectively improve the reasoning ability of the model. DCMN+A$_{\text{OI}}$ works better than DCMN on *detail* and *main* categories which indicates that the model achieves better distinguish ability with answer option interaction information.

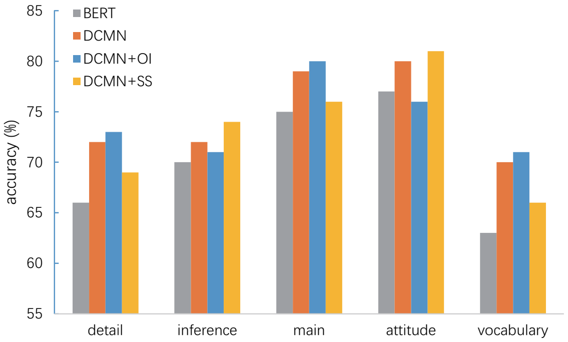

Figure 3: Results on different question types, tested on the RACE dev set. BERT$_{\text{large}}$ is used as encoder here. OI: Answer Option Interaction. SS: Passage Sentence Selection.

Related Work

Neural network based methods have been applied to several natural language processing tasks, especially to MRC (Zhang et al. 2019c; Zhang, Huang, and Zhao 2018).

The task of selecting sentences to answer the question has been studied across several question-answering (QA) datasets, by modeling the relevance between a sentence and the question (Min et al. 2018b; Wang et al. 2019; Choi et al. 2017; Raiman and Miller 2017; Wang et al. 2018a). (Wang et al. 2019) apply distant supervision to generate imperfect labels and then use them to train a neural evidence extractor. (Min et al. 2018b) propose a simple sentence selector to select the minimal set of sentences then feed into the QA model. They are different from our work in that (i) we select the sentences by modeling the relevance among sentence-question-option triplet, not sentence-question pair. (ii) Our model uses the output of language model as the sentence embedding and computes the relevance score using these sentence vectors directly, without the need of manually defined labels. (iii) We achieve a generally positive impact by selecting sentences while previous sentence selection methods usually bring performance decrease in most cases.

Most recent works attempting to integrate answer option interaction information focus on building attention mechanism at word-level (Ran et al. 2019; Zhu et al. 2018; Pujari and Goldwasser 2019) whose performance increase is very limited. Our answer option interaction module is different from previous works in that: (i) we encode the comparison information by modeling the bilinear representation among the options at sentence-level which is similar to modeling passage-question sequence relationship, other than attention mechanism. (ii) We use gated mechanism to fuse the comparison information into the original answer option representations.

Conclusion

This paper proposes dual co-matching network integrated with two reading strategies (passage sentence selection and answer option interaction) to enhance multi-choice machine reading comprehension. In terms of strong pre-trained language models such as BERT and XLNet as encoder, our proposed method achieves state-of-the-art results on five representative multi-choice MRC datasets including RACE. The experiment results consistently indicate the general effectiveness and applicability of our model.

References

Chen, Z.; Cui, Y.; Ma, W.; and Wang, S. 2019. Convolutional Spatial Attention Model for Reading Comprehension with Multiple-Choice Questions. In *AAAI 2019*.

Choi, E.; Hewlett, D.; Uszkoreit, J.; Polosukhin, I.; Lacoste, A.; and Berant, J. 2017. Coarse-to-Fine Question Answering for Long Documents. In *ACL 2017*, 209–220.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL 2019*, 4171–4186.

Hermann, K. M.; Kocisky, T.; Grefenstette, E.; Espeholt, L.; Kay, W.; Suleyman, M.; and Blunsom, P. 2015. Teaching Machines to Read and Comprehend. In *NIPS 2015*.

Lai, G.; Xie, Q.; Liu, H.; Yang, Y.; and Hovy, E. 2017. RACE: Large-scale ReAding Comprehension Dataset From Examinations. In *EMNLP 2017*, 785–794.

Li, Z.; Ding, X.; and Liu, T. 2019. Story Ending Prediction by Transferable BERT. In *IJCAI-19*, 1800–1806.

Luo, Y.; Xiao, F.; and Zhao, H. 2019. Hierarchical Contextualized Representation for Named Entity Recognition. In *AAAI 2020*.

Min, S.; Zhong, V.; Socher, R.; and Xiong, C. 2018a. Efficient and Robust Question Answering from Minimal Context over Documents. In *ACL 2018*, 1725–1735.

Min, S.; Zhong, V.; Socher, R.; and Xiong, C. 2018b. Efficient and Robust Question Answering from Minimal Context over Documents. In *ACL 2018*, 1725–1735.

Mostafazadeh, N.; Chambers, N.; He, X.; Parikh, D.; Batra, D.; Vanderwende, L.; Kohli, P.; and Allen, J. 2016. A Corpus and Cloze Evaluation for Deeper Understanding of Commonsense Stories. In *NAACL 2016*, 839–849.

Nguyen, T.; Rosenberg, M.; Song, X.; Gao, J.; Tiwary, S.; Majumder, R.; and Deng, L. 2016. MS MARCO: A Human Generated MAchine Reading COmprehension Dataset. *CoRR* abs/1611.09268.

Ostermann, S.; Roth, M.; Modi, A.; Thater, S.; and Pinkal, M. 2018. SemEval-2018 Task 11: Machine Comprehension Using Commonsense Knowledge. In *Proceedings of The 12th International Workshop on Semantic Evaluation*.

Pujari, R., and Goldwasser, D. 2019. Using Natural Language Relations between Answer Choices for Machine Comprehension. In *NAACL-HLT 2019*, 4010–4015.

Qian, D., and Schedl, M. 2004. Evaluation of an In-Depth Vocabulary Knowledge Measure for Assessing Reading Performance. *Language Testing - LANG TEST* 21:28–52.

Radford, A. 2018. Improving Language Understanding by Generative Pre-Training. In *OpenAI preprint*.

Raiman, J., and Miller, J. 2017. Globally Normalized Reader. In *EMNLP 2017*, 1059–1069.

Rajpurkar, P.; Zhang, J.; Lopyrev, K.; and Liang, P. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *EMNLP 2016*, 2383–2392.

Ran, Q.; Li, P.; Hu, W.; and Zhou, J. 2019. Option Comparison Network for Multiple-choice Reading Comprehension. *CoRR* abs/1903.03033.

Richardson, M.; Burges, C. J.; and Renshaw, E. 2013. MCTest: A Challenge Dataset for the Open-Domain Machine Comprehension of Text. In *EMNLP 2013*, 193–203.

Srivastava, R. K.; Greff, K.; and Schmidhuber, J. 2015. Highway networks. In *ICML 2015*.

Sun, K.; Yu, D.; Yu, D.; and Cardie, C. 2019. Improving Machine Reading Comprehension with General Reading Strategies. In *NAACL 2019*.

Tang, M.; Cai, J.; and Zhuo, H. H. 2019. Multi-Matching Network for Multiple Choice Reading Comprehension. In *AAAI 2019*.

Tay, Y.; Tuan, L. A.; and Hui, S. C. 2018. Multi-range Reasoning for Machine Comprehension. *CoRR* abs/1803.09074.

Trischler, A.; Wang, T.; Yuan, X.; Harris, J.; Sordoni, A.; Bachman, P.; and Suleman, K. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, 191–200.

Wang, L.; Sun, M.; Zhao, W.; Shen, K.; and Liu, J. 2018a. Yuanfudao at SemEval-2018 Task 11: Three-way Attention and Relational Knowledge for Commonsense Machine Comprehension. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, 758–762.

Wang, S.; Yu, M.; Jiang, J.; and Chang, S. 2018b. A Co-Matching Model for Multi-choice Reading Comprehension. In *ACL 2018*, 746–751.

Wang, H.; Yu, D.; Sun, K.; Chen, J.; Yu, D.; Roth, D.; and McAllester, D. A. 2019. Evidence Sentence Extraction for Machine Reading Comprehension. *CoRR* abs/1902.08852.

Xiao, F.; Li, J.; Zhao, H.; Wang, R.; and Chen, K. 2019. Lattice-Based Transformer Encoder for Neural Machine Translation. In *ACL 2019*, 3090–3097.

Yang, Z.; Dai, Z.; Yang, Y.; Carbonell, J. G.; Salakhutdinov, R.; and Le, Q. V. 2019. XLNet: Generalized Autoregressive Pretraining for Language Understanding. In *NIPS 2019*.

Zhang, Z.; Huang, Y.; Zhu, P.; and Zhao, H. 2018. Effective Character-augmented Word Embedding for Machine Reading Comprehension. In *NLPCC 2018*.

Zhang, Z.; Wu, Y.; Zhao, H.; Li, Z.; Zhang, S.; Zhou, X.; and Zhou, X. 2019a. Semantics-aware BERT for Language Understanding. In *AAAI 2020*.

Zhang, Z.; Wu, Y.; Zhou, J.; Duan, S.; Zhao, H.; and Wang, R. 2019b. SG-Net: Syntax-Guided Machine Reading Comprehension. In *AAAI 2020*.

Zhang, Z.; Zhao, H.; Ling, K.; Li, J.; Li, Z.; and He, S. 2019c. Effective Subword Segmentation for Text Comprehension. In *IEEE/ACM Transactions on Audio, Speech, and Language Processing, 2019*.

Zhang, Z.; Huang, Y.; and Zhao, H. 2018. Subword-augmented Embedding for Cloze Reading Comprehension. In *COLING 2018*, 1802–1814.

Zhou, J., and Zhao, H. Head-Driven Phrase Structure Grammar Parsing on Penn Treebank. In *ACL 2019*, 2396–2408.

Zhu, H.; Wei, F.; Qin, B.; and Liu, T. 2018. Hierarchical Attention Flow for Multiple-choice Reading Comprehension. In *AAAI 2018*.