REPT: Bridging Language Models and Machine Reading Comprehension via Retrieval-Based Pre-training

Fangkai Jiao[1]*, **Yangyang Guo**[1], **Yilin Niu**[2], **Feng Ji**[3], **Feng-Lin Li**[3], **Liqiang Nie**[1]†
[1] School of Computer Science and Technology, Shandong University, Qingdao, China
[2] Department of Computer Science and Technology, Tsinghua University, Beijing, China
[3] Damo Academy, Alibaba Group, Hangzhou, China
jiaofangkai@hotmail.com guoyang.eric@gmail.com niuyl14j@gmail.com
{zhongxiu.jf, fenglin.lfl}@alibaba-inc.com nieliqiang@gmail.com

Abstract

Pre-trained Language Models (PLMs) have achieved great success on Machine Reading Comprehension (MRC) over the past few years. Although the general language representation learned from large-scale corpora does benefit MRC, the poor support in evidence extraction which requires reasoning across multiple sentences hinders PLMs from further advancing MRC. To bridge the gap between general PLMs and MRC, we present REPT, a REtrieval-based Pre-Training approach. In particular, we introduce two self-supervised tasks to strengthen evidence extraction during pre-training, which is further inherited by downstream MRC tasks through the consistent retrieval operation and model architecture. To evaluate our proposed method, we conduct extensive experiments on five MRC datasets that require collecting evidence from and reasoning across multiple sentences. Experimental results demonstrate the effectiveness of our pre-training approach. Moreover, further analysis shows that our approach is able to enhance the capacity of evidence extraction without explicit supervision.

1 Introduction

Machine Reading Comprehension (MRC) is an important task to evaluate the machine understanding of natural language. Given a set of documents and a question (with possible options), an MRC system is required to provide the correct answer by either retrieving a meaningful span (Rajpurkar et al., 2018a) or selecting the correct option from a few candidates (Lai et al., 2017; Sun et al., 2019; Guo et al., 2019, 2021). Recently, with the development of self-supervised learning, the pre-trained language models (Devlin et al., 2019; Yang et al., 2019b) fine-tuned on several machine reading comprehension benchmarks (Reddy et al., 2019; Kwiatkowski et al., 2019) have achieved superior performance. The dominant reason lies in the strong and general contextual representation learned from large-scale natural language corpora. Nevertheless, PLMs focus more on the general language representation and semantics to benefit various downstream tasks, while MRC demands the capability of extracting evidence across one or multiple documents and performing reasoning over the collected clues (Fang et al., 2020; Yang et al., 2018). Put it differently, there exists an obvious gap, indicating an insufficient exploitation of PLMs over MRC.

Some efforts have been made to bridge the gap between PLMs and downstream tasks, which can be roughly divided into two categories: knowledge enhancement and task-oriented pre-training (Qiu et al., 2020). The former introduces commonsense or world knowledge into the pre-training (Zhang et al., 2019; Sun et al., 2020; Varkel and Globerson, 2020; Ye et al., 2020) or fine-tuning (Yang et al., 2019a) for better performance over knowledge-driven tasks. And the latter includes some delicately designed pre-training tasks, e.g., the contrastive approach of learning discourse knowledge towards textual entailment task (Iter et al., 2020). Although these approaches have achieved some improvements on certain tasks, few of them are specifically designed for evidence extraction, which is indeed indispensable to MRC.

In fact, equipping PLMs with the capability of evidence extraction in MRC is challenging due to the following two factors. 1) The process of collecting clues from a document is difficult to be integrated into PLMs without designing specific model architectures or pre-training tasks (Qiu et al., 2020; Zhao et al., 2020). And 2) large-scale pre-training process would make PLMs overfit to pre-

*Work is done during internship at Alibaba Group.
†Corresponding author: Liqiang Nie.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 150–163

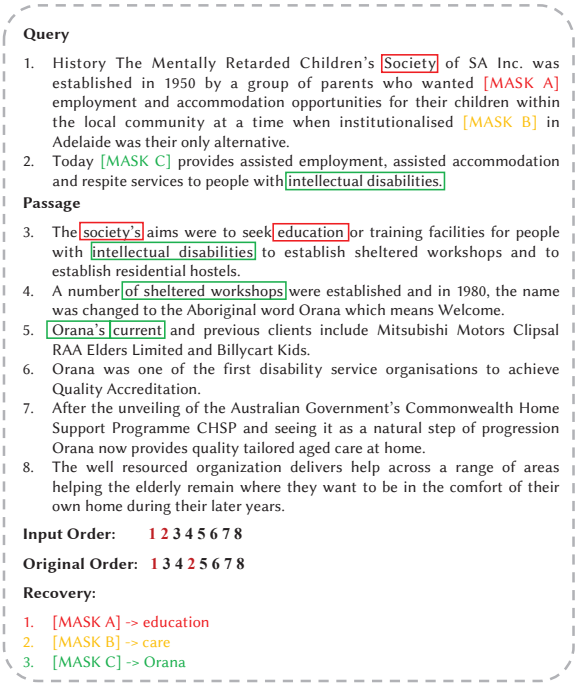


Figure 1: A running example obtained from our method. The query sentences are extracted from the original document with some crucial information being randomly masked, i.e., the sentence 1 and 2. The model is required to predict the preceding and following sentence for each query in the original document and recover the masked clues, i.e., infer the original order from input order and fill the [MASK] with the initial token. The phrases in boxes are the possible clues for recovering the masked tokens and the correct order.

training tasks (Chung et al., 2021; Tamkin et al., 2020). In other words, it is difficult to take full advantage of the pre-training merits if the training objectives of pre-training and downstream MRC are greatly separated.

To deal with the aforementioned challenges, we propose a novel retrieval-based pre-training approach, REPT, to bridge the gap between PLMs and MRC. Firstly, to unify the training objective, we design a novel pre-training task, namely Surrounding Sentences Prediction (SSP), as illustrated in Figure 1. Given a document, several sentences will be firstly selected as queries, and the others are jointly treated as a passage[1]. Thereafter, for each query, the model should predict its preceding and following sentences in the original document by collecting clues from each sentence, which is compatible with evidence extraction in MRC tasks. It is worth emphasizing that, the repeated occurrence of entities or nouns across different sentences often lead to information short-cut (Lee et al., 2020), from which the order of sentences can be easily recovered. In view of this, we propose to mask such explicit clues. As a result, the model is enforced to infer the correct positions of queries by gathering evidence with the incomplete information. Secondly, to preserve the effectiveness of contextual representation, the masked clues are also required to be recovered through retrieving relevant information from other parts of the document, which is implemented via our Retrieval based Masked Language Modeling (RMLM) task.

In this way, the pre-training stage can be properly aligned with MRC: 1) the training objectives are connected through the introduction of the two pre-training tasks, which will be inherited by downstream MRC tasks through consistent retrieval operation. And 2) the capability of evidence extraction from documents or sentences is enhanced during pre-training, and will be smoothly transferred to MRC. Our contributions in this paper are summarized as follows:

1. We present REPT, a novel pre-training approach, to bridge the gap between PLMs and MRC through retrieval-based pre-training.

2. We design two self-supervised pre-training tasks, i.e., SSP and RMLM, to augment PLMs with the ability of evidence extraction with the help of retrieval operation and eliminating information short-cut, which can be smoothly transferred to downstream MRC tasks.

3. We evaluate our method over five reading comprehension benchmarks of two different task forms: Multiple Choice QA (MCQA) and Span Extraction (SE). The substantial improvements over strong baselines demonstrate the effectiveness of our pre-training approach. We conduct an empirical study to verify that our method are able to enhance evidence extraction as expected.

2 Related Work

MRC has received increasing attention in recent years. Many challenging benchmarks have been established to examine various forms of reasoning abilities, e.g., multi-hop (Yang et al., 2018), discrete (Dua et al., 2019), and logic reasoning (Yu et al., 2020). To solve the problem, a typical design is to gather possible clues through entity linking

[1]We use *passage* here to keep consistent with MRC tasks. And *document* refers to the combination of queries and *passage*.

(Zhao et al., 2020) or self-constructed graph (Fang et al., 2020; Ran et al., 2019), and then perform multi-step reasoning. It is worth noting that, gathering clues is vital but challenging, especially for long document understanding. Some efforts have been dedicated to improving evidence extraction via direct (Wang et al., 2018) or distant supervision (Niu et al., 2020).

Generally, the fine-tuned PLMs (Devlin et al., 2019; Yang et al., 2019b) can obtain superior performance in MRC due to their strong and general language representation. However, there still exist some gaps between PLMs and various downstream tasks, since certain abilities required by the downstream tasks cannot be learned through the existing pre-training tasks (Qiu et al., 2020). In order to take full advantage of PLMs, a few studies attempt to align the pre-training and fine-tuning stages. For example, Tamborrino et al. (2020) reformulated the commonsense question answering task as scoring via leveraging the predicted probabilities for Masked Language Modeling (MLM) in RoBERTa (Liu et al., 2019). With the help of the commonsense learned through MLM, the method achieves comparable results with supervised approaches in zero-shot setting, indicating that bridging the gap between these two stages yields considerable improvement. Chung et al. (2021) tried to address the overfitting problem during pre-training through decoupling input and output embedding weights and enlarging the embedding size during decoding. The resultant model is therefore more transferable across tasks and languages.

In addition, some task-oriented pre-training methods have also been developed. For instance, Wang et al. (2020) proposed a novel pre-training method for sentence representation learning, where the masked tokens in a sentence are forced to be recovered from other sentences through sentence-level attention. Based on this, the attention weights can be directly fine-tuned to rank the candidates in answer selection or information retrieval. Lee et al. (2019) tried to learn the dense document representation for information retrieval by minimizing the distance between the representation of an query sentence and its context. Guu et al. (2020) designed an augmented MLM tasks to jointly train a neural retriever and a language model for Open-domain QA. Different from these methods ranking the documents for open-domain QA, our approach focuses on enhancing the ability of evidence extraction in MRC, where the MLM based task by it alone is insufficient.

3 Method

In this section, we present the details of the proposed method, REPT. We firstly describe the data pre-processing part (§3.1), and then illustrate the two pre-training tasks, i.e., SSP and RMLM (§3.3) and the training objectives (§3.4). Finally, we detail how to fine-tune our pre-trained model for downstream tasks through retrieval-based evidence extraction (§3.5).

3.1 Data Pre-processing

For pre-training, we use the English Wikipedia[2] as our training data. We divide each Wikipedia article into segments, each containing up to 500 tokens[3] without overlapping. We treat each segment as a document and split it into several sentences[4].

In order to increase the difficulty and efficiency of pre-training, for each document, we select 30% of the most important sentences as queries and the rest in their original order as a passage. Specifically, the importance of each sentence in a document is measured through the summation of the importance of entities and nouns it contains, which is further defined as the number of sentences an entity/noun occurs. Hereafter, masking is introduced to entities and nouns in queries according to pre-defined ratios to eliminate information short-cut. More details about the masking strategy are described in Appendix A and an example after pre-processing can be found in Figure 1.

3.2 Task Definition

We treat a document as a sequence of n sequential sentences with m tokens. Supposing that there are t sentences selected as queries following §3.1, the rearranged sequence is defined as $\mathcal{S} = [\mathbf{s}^1, \mathbf{s}^2, \cdots, \mathbf{s}^t, \cdots, \mathbf{s}^n]$, and the index of queries is $\mathcal{Q} = \{1, 2, \cdots, t\}$. Besides, we define a mapping function r to map the rearranged sentences to their original position. Taking Figure 1 as an example, the mapping $r(\mathbf{s}^1) = 1,\ r(\mathbf{s}^2) = 4,\ r(\mathbf{s}^3) = 2$ and $(\mathbf{s}^4) = 3$ indicates that the original order is $\{\mathbf{s}^1, \mathbf{s}^3, \mathbf{s}^4, \mathbf{s}^2, \cdots\}$.

Taking $\mathcal{S}$ as input, the Surrounding Sentences Prediction task should predict the correct sentence

[2]We use the 2020/05/01 dump.

[3]The tokenized sub-words following BERT and RoBERTa.

[4]Any sentence with less than five tokens is concatenated to its previous one.

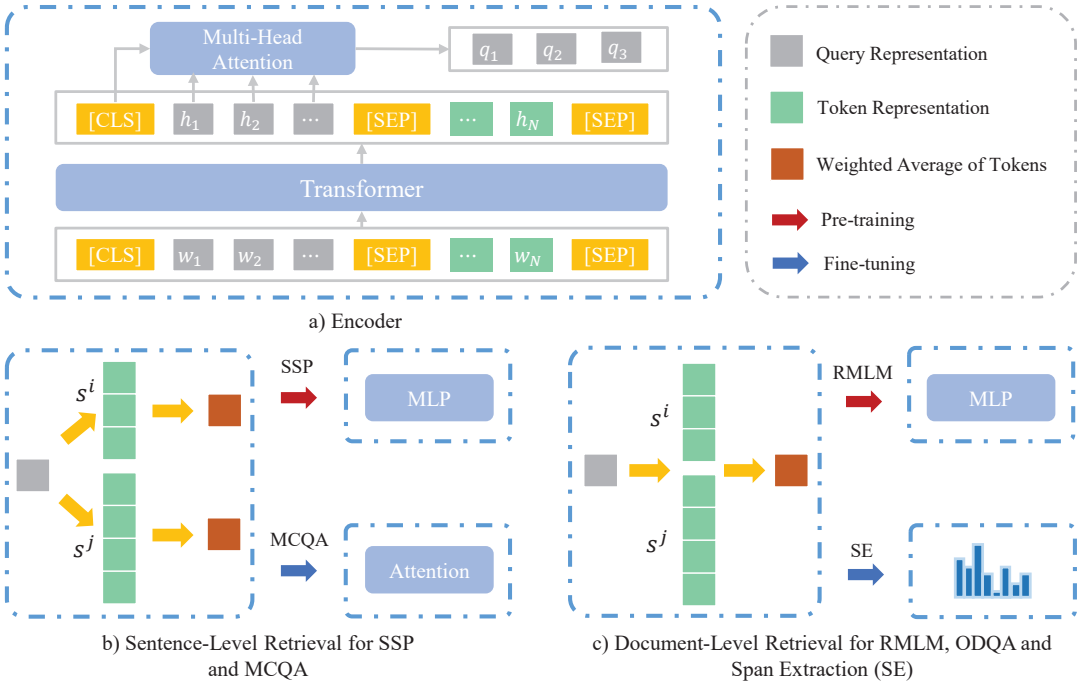


Figure 2: Framework of our model. a) Encoder composed of a pre-trained Transformer encoder and a query generator based on multi-head attention. b) The attention-based sentence-level retrieval for evidence extraction for each sentence, which will be further adopted by SSP during pre-training and MCQA during fine-tuning. c) The attention-based document-level retrieval for evidence extraction among the input sequence, which is employed for RMLM. For SE, the similarity function is directly fine-tuned.

index a and b for each query $\mathbf{s}^q$ with $q \in \mathcal{Q}$[5]:

$$\begin{cases} r(\mathbf{s}^a) = r(\mathbf{s}^q) - 1, \\ r(\mathbf{s}^b) = r(\mathbf{s}^q) + 1. \end{cases} \quad (1)$$

As for the Retrieval based Masked Language Modeling (RMLM) task, the model should recover all the masked tokens in each query $\mathbf{s}^q$.

3.3 Model

First of all, we leverage a pre-trained Transformer (Vaswani et al., 2017), such as BERT, as our encoder to obtain the contextual representation of sentences. The output of Transformer is formulated as:

$$\mathbf{H} = [\mathbf{h}_{\text{cls}}, \cdots, \mathbf{h}_m, \mathbf{h}_{\text{sep}}] = \text{Encoder}(\tilde{\mathcal{S}}), \quad (2)$$

where $\mathbf{H} \in \mathbb{R}^{d \times (m+3)}$, and d is the hidden size. For a better illustration, we will use $\mathbf{H}^i$ to represent the hidden state matrix of tokens that belong to sentence $\mathbf{s}^i$, such that:

$$\mathbf{H} = [\mathbf{H}^1, \mathbf{H}^2, \cdots, \mathbf{H}^n], \ \mathbf{H}^i \in \mathbb{R}^{d \times l_i},$$

where l_i is the length of sentence $\mathbf{s}^i$ and $m = \sum_i l_i$. Since the process for each query is exactly the same, we use $q \in \mathcal{Q}$ as a representative to introduce the calculation with respect to each query below.

[5]Specifically, for $r(\mathbf{s}^q) = 1$ or $r(\mathbf{s}^q) = n$, the corresponding prediction task is removed since its preceding or following sentence does not exist.

3.3.1 Query Representation

In order to gather potential clues from a document or sentences, we adopt the multi-head attention mechanism proposed by (Vaswani et al., 2017) to obtain the sentence-level representation for each query. Formally, the attention mechanism is defined as $\text{MHA}(\mathbf{Q}, \mathbf{K}, \mathbf{V})$, where $\mathbf{Q}, \mathbf{K}, \mathbf{V}$ are query, key and value matrices, respectively. To consider the global information, we leverage $\mathbf{h}_{\text{cls}}$ as the query vector, and $\mathbf{H}^q$ as $\mathbf{K}$ and $\mathbf{V}$:

$$\mathbf{v}_0^{q\top} = \text{MHA}(\mathbf{h}_{\text{cls}}^\top, \mathbf{H}^q, \mathbf{H}^q). \quad (3)$$

During pre-training, we reuse the layer defined by Equation 3 with $\mathbf{Q} = \mathbf{v}_0^q$ and $\mathbf{K} = \mathbf{V} = \mathbf{H}^q$, to generate the task-specific query representation $\mathbf{v}^q$, which is designed to alleviate the overfitting problem (He et al., 2021).

3.3.2 Surrounding Sentence Prediction

To enhance the capability of pre-trained models for evidence extraction, we have carefully designed the SSP task, where the model should predict the preceding and following sentences for a given query by extracting the relevant evidence from each sentence. Consequently, we introduce a retrieval operation, which is implemented via a single-head attention

mechanism[6]:

$$\mathbf{u}_q^{i\top} = \mathrm{Att}(\mathbf{v}^{q\top}, \mathbf{H}^i, \mathbf{H}^i), \tag{4}$$

where $\mathbf{u}_q^i$ is the representation of sentence $\mathbf{s}^i$, highlighting the evidence information pertaining to query $\mathbf{s}^q$. Finally, the score of each sentence in the document with regard to $\mathbf{s}^q$ is obtained through:

$$\mathbf{o}_q^i = \mathbf{W}_2(\tanh(\mathbf{W}_1\mathbf{u}_q^i + \mathbf{b}_1)) + \mathbf{b}_2. \tag{5}$$

3.3.3 Retrieval based MLM

Since the masking noise introduced when constructing queries could also bring inconsistency between pre-training and fine-tuning, we further designed a retrieval based MLM task to alleviate this problem. In the RMLM task, the model should predict the masked entities or nouns through retrieving relevant information from a document. More specifically, the query-aware evidence representation of the input sequence is obtained via:

$$\mathbf{g}^{q\top} = \mathrm{Att}(\mathbf{v}^{q\top}, \mathbf{H}, \mathbf{H}). \tag{6}$$

Denoting the index of a masked token in query $\mathbf{s}^q$ as z, the representation of the masked token s_z^q used for recovering is:

$$\tilde{\mathbf{h}}_z^q = f(\mathbf{h}_z, \mathbf{g}^q), \tag{7}$$

where the function $f(\cdot,\cdot)$ is implemented as a normalized 2-layer feed-forward network, and the details are illustrated in Appendix B.2.

3.4 Optimization

As the definition in Equation 1, given a and b as the index of the original preceding and following sentences of the query $\mathbf{s}^q$ in $\mathcal{S}$, the corresponding probabilities for surrounding sentences are formulated as:

$$\begin{cases} p_{\mathrm{ssp}}(a|q,\mathcal{S}) = \dfrac{\exp(\mathbf{o}_q^a)}{\sum_{j=1,j\notin\{b,q\}}^{n}\exp(\mathbf{o}_q^j)}, \\ p_{\mathrm{ssp}}(b|q,\mathcal{S}) = \dfrac{\exp(\mathbf{o}_q^b)}{\sum_{j=1,j\notin\{a,q\}}^{n}\exp(\mathbf{o}_q^j)}. \end{cases} \tag{8}$$

The objective of SSP is subsequently defined as:

$$\mathcal{L}_{\mathrm{ssp}} = \mathbb{E}(-\frac{1}{|\mathcal{Q}|}\sum_q(\log p_{\mathrm{ssp}}(a|q,\mathcal{S}) + \log p_{\mathrm{ssp}}(b|q,\mathcal{S}))). \tag{9}$$

[6]The details are illustrated in Appendix B.1.

As for RMLM, supposing the index set of masked tokens in query $\mathbf{s}^q$ is $\mathcal{Z}^q$, and the set of corresponding original tokens is $\mathcal{X}^q$, the probability for recovering a masked token is:

$$p_{\mathrm{rmlm}}(x_z|z,q,\mathcal{S}) = \frac{\exp(\mathrm{e}(x_z)^\top\tilde{\mathbf{h}}_z^q)}{\sum_{x'}\exp(\mathrm{e}(x')^\top\tilde{\mathbf{h}}_z^q)}, \tag{10}$$

where $z \in \mathcal{Z}^q, x_z \in \mathcal{X}^q$, x' is a token in vocabulary, and $\mathrm{e}(x)$ denotes the word embedding of x. Then the objective of RMLM is:

$$\mathcal{L}_{\mathrm{rmlm}} = \mathbb{E}(-\frac{\sum_q\sum_z\log p_{\mathrm{pmlm}}(x_z|z,q,\mathcal{S})}{\sum_q|\mathcal{Z}^q|}). \tag{11}$$

During pre-training, the model tries to optimize the two objectives jointly:

$$\mathcal{L} = \mathcal{L}_{\mathrm{ssp}} + \mathcal{L}_{\mathrm{rmlm}}. \tag{12}$$

3.5 Fine-tuning

During fine-tuning, the input contains a query sentence and a passage. For multiple choice QA tasks, we concatenate a question with an option to form a question-option pair and use it as a whole query. In this section, we use $q = 0$ to represent the index of the query and the sentences of passage are kept in their original order. The input sequence can be thus denoted as:

$$\mathcal{S} = [s^q, s^1, s^2, \cdots, s^n].$$

To inherit the evidence extraction ability augmented during pre-training, we incorporate the same retrieval operation into fine-tuning to collect clues from the passage. Firstly, we reuse the attention mechanism defined in Equation 3 to obtain the query representation $\mathbf{v}^q$. As for the evidence extraction process, we formulate it differently for Multiple Choice QA and Span Extraction.

3.5.1 Multiple Choice QA

Similar to Equation 4, we adopt an attention mechanism, whereby the query-aware sentence representation $\mathbf{u}_q^i$ is obtained via gathering evidence from each sentence:

$$\mathbf{u}_q^{i\top} = \mathrm{Att}(\mathbf{v}^{q\top}, \mathbf{H}^i, \mathbf{H}^i),\ i \neq q. \tag{13}$$

And the final passage representation highlighting the evidence can be obtained via the sentence-level evidence extraction:

$$\mathbf{v}^p = \mathrm{Att}(\mathbf{v}^{q\top}, \mathbf{U}, \mathbf{U}), \tag{14}$$

where $\mathbf{U} = [\mathbf{u}_q^1, \cdots, \mathbf{u}_q^n]$ and $\mathbf{U} \in \mathbb{R}^{d \times n}$. Finally, we represent the probability of each option c using both the query $\mathbf{v}^q$ and the passage $\mathbf{v}^p$:

$$p_c^{\text{mc}} \propto \exp(\mathbf{W}_6(\tanh(\mathbf{W}_5[\mathbf{v}^q; \mathbf{v}^p] + \mathbf{b}_5)) + \mathbf{b}_6). \tag{15}$$

Specifically, for Multi-RC, since the number of correct answer options for each question is uncertain, the task is often treated as a binary classification problem for each option. As a result, we adopt a MLP to get the probability of whether an option c is correct:

$$p_c^{\text{mc}} = \sigma(\mathbf{W}_8(\tanh(\mathbf{W}_7[\mathbf{v}^q; \mathbf{v}^p] + \mathbf{b}_7)) + \mathbf{b}_8), \tag{16}$$

where σ is the sigmoid function.

3.5.2 Span Extraction

Since answer spans are often consistent with corresponding evidences, we directly leverage the query to extract relevant spans. The probability of selecting start position s and end position e of an answer span is given by:

$$\begin{cases} p_s^{\text{span}} \propto \exp(\mathbf{v}^{q\top} \mathbf{W}_9 \mathbf{h}_s), \\ p_e^{\text{span}} \propto \exp(\mathbf{v}^{q\top} \mathbf{W}_{10} \mathbf{h}_e). \end{cases} \tag{17}$$

4 Experiment

4.1 Dataset

4.1.1 Multiple Choice Question Answering

DREAM (Sun et al., 2019) contains 10,197 multiple choice questions for 6,444 dialogues collected from English Examinations designed by human experts, in which 85% of the questions require reasoning across multiple sentences, and 34% of the questions also involve commonsense knowledge.

RACE (Lai et al., 2017) is a large-scale reading comprehension dataset collected from English Examinations and created by domain experts to test students' reading comprehension skills. It has a wide variety of question types, e.g., summarization, inference, deduction and context matching, and requires complex reasoning techniques.

Multi-RC (Khashabi et al., 2018) is a dataset of short paragraphs and multi-sentence questions. The number of correct answer options for each question is not pre-specified and the correct answer(s) is not required to be a span in the text. Moreover, the dataset provides annotated evidence sentence.

ReClor (Yu et al., 2020) is extracted from logical reasoning questions of standardized graduate admission examinations. Existing studies show that the state-of-the-art models perform poorly on ReClor, indicating the deficiency of logical reasoning ability of current PLMs.

4.1.2 Span Extraction

Hotpot QA (Yang et al., 2018) is a question answering dataset involving natural and multi-hop questions. The challenge contains two settings, the distractor setting and the full-wiki setting. In this paper, we focused on the full-wiki setting, where the system should retrieve the relevant paragraphs from Wikipedia and then predict the answer.

SQuAD2.0 (Rajpurkar et al., 2018b) is reading comprehension dataset, consisting of questions posed by crowdworkers on a set of Wikipedia articles, where the answer to every question is a segment of text, or span, from the corresponding reading passage, or the question might be unanswerable.

4.2 Implementation Detail

We leave the details about the implementation and pre-training corpora in Appendix A due to the limitation of space.

4.3 Baseline

Since our method is used for further pre-training, we mainly compared our model with BERT/RoBERTa and their variants. For Hotpot QA, we integrated our models into an open-sourced and well-accepted system (Asai et al., 2020) and evaluated the performance. The details of baselines are summarized as follows:

4.3.1 Multiple Choice QA

BERT is the BERT-base model with 2-layer MLP as the task-specific module.

BERT-Q & **RoBERTa-Q** refer to the designed but not further trained models, which include an extra multi-head attention for generating query representation via Equation 3, and our retrieval operation for evidence extraction as in §3.5.1 and §3.5.2.

BERT-Q w. R/S & **RoBERTa-Q w. R/S** refer to the designed models further trained with our proposed SSP and RMLM tasks (denoted as **S** and **R**, respectively).

BERT-Q w. R & **BERT-Q w. S** refer to the models further trained with only one pre-training task, RMLM or SSP.

BERT-Q w. M & **BERT w. M** refer to the models further trained with MLM. For fair comparison, we

Model / Dataset	RACE		DREAM		ReClor		Multi-RC		
	Dev Acc.	Test Acc.	Dev Acc.	Test Acc.	Dev Acc.	Test Acc.	Dev EM	Dev $F1_a$	Dev $F1_m$
BERT-base†	–	65.0	63.4	63.2	**54.6**	47.3	–	–	–
BERT w. M	67.7	66.3	62.9	63.2	51.6	45.1	26.6	71.8	74.2
BERT-Q	67.2	65.2	62.9	62.3	48.4	45.0	22.8	69.6	72.0
BERT-Q w. M	67.7	66.9	61.8	62.2	48.8	48.3	23.8	70.1	72.6
BERT-Q w. R	65.5	64.7	59.0	58.6	46.8	45.1	26.4	71.5	74.0
BERT-Q w. S	69.5	66.5	**64.8**	62.2	52.0	46.5	30.0	73.0	75.8
BERT-Q w. R/S	**70.1**	**68.1**	64.4	**64.0**	50.6	**49.2**	**31.9**	**73.8**	**76.3**
RoBERTa-base	76.0	75.5	**71.2**	69.8	54.8	**50.8**	38.7	77.1	79.2
RoBERTa-Q	76.8	**75.7**	70.9	69.5	**56.0**	49.7	34.6	75.4	77.4
RoBERTa-Q w. R/S	**77.1**	74.9	70.9	**70.8**	54.8	50.3	**40.4**	**77.6**	**80.0**

Table 1: Results on multiple choice question answering tasks. ($F1_a$: F1 score on all answer-options; $F1_m$: macro-average F1 score of all questions.) We ran all experiments using **four** different random seeds with the same hyper-parameters, and report the average performance, except for ReClor and Multi-RC. For ReClor, we submitted the best model on the development set to the leaderboard to get the results on the test set. For MultiRC, we merely reported the performance on development set since the test set is unavailable. †: The results are reported by the leaderboard.

further train BERT with the same Wikipedia corpus for equivalent steps.

4.3.2 Hotpot QA

For hotpot QA, we constructed the system based on Graph-based Recurrent Retriever (Asai et al., 2020), which includes a retriever and a reader based on BERT. We simply replaced the reader with our models and evaluated their performance in comparison with several published strong baselines on the leaderboard[7].

5 Results and Analyses

5.1 Results for Multiple Choice QA

Table 1 shows the results of the baselines and our method on multiple choice question answering.

From Table 1, we can observe that: 1) Compared with BERT-Q and BERT, our method significantly improves the performance over all the datasets, which validates the effectiveness of our proposed pre-training method. 2) As for the model structure, BERT-Q obtains similar or worse results compared with BERT, which suggests that the retrieval operation can hardly improve the performance without specialised pre-training. 3) Taking the rows of BERT, BERT-Q, BERT w. M, BERT-Q w. M for comparison, the models with further pre-training using MLM achieve similar or slightly higher performance. The results show that further training BERT using MLM and the same corpus can only achieve very limited improvements. 4) Regarding the two pre-training tasks, BERT-Q w. R/S leads to similar performance on the development sets compared with BERT-Q w. S, but a much higher accuracy on the test sets, which suggests RMLM can help to maintain the effectiveness of contextual language representation. However, there is a significant degradation over all datasets for BERT-Q w. R. The main reason is possibly because the model cannot tolerate the sentence shuffling noise, which may lead to the discrepancy between pre-training and MRC, and thus need to be alleviated through SSP. And 5) considering the experiments over RoBERTa-based models, RoBERTa-Q w. R/S outperforms RoBERTa-Q and RoBERTa-base with considerable improvements over Multi-RC and the test set of DREAM, which also indicates that our method can benefit stronger PLMs.

5.2 Performance on Span Extraction QA

The results of span extraction on Hotpot QA are shown in Table 2. We constructed the system using the Graph Recurrent Retriever (GRR) proposed by Asai et al. (2020) and different readers. As shown in the table, GRR + BERT-Q w. R/S outpeforms GRR + BERT-base by more than 2.5% absolute points on both EM and F1. And GRR + RoBERTa-Q w. R/S also achieves a significant improvement over GRR + RoBERTa-base. During the test stage, our best system, GRR + RoBERTa-Q w. R/S performs better than the strong baselines and get closer to GRR + BERT-wwm-large. The above results strongly demonstrate the effectiveness of our pre-

[7] https://hotpotqa.github.io/.

Model / Dataset	Dev		Test	
	EM	F1	EM	F1
Transformer-XH (Zhao et al., 2020)	54.0	66.2	51.6	64.7
HGN (Fang et al., 2020)	–	–	56.7	69.2
GRR + BERT-wwm-Large*	**60.5**	**73.3**	**60.0**	**73.0**
GRR + BERT-base*	52.7	65.8	–	–
GRR + BERT-Q w. R/S	**55.2**	**68.4**	–	–
GRR + RoBERTa-base	56.8	69.6	–	–
GRR + RoBERTa-Q w. R/S	**58.4**	**71.3**	58.1	71.0

Table 2: Results of our method and other strong baselines on Hotpot QA. *GRR* means the Graph Recurrent Retriever proposed by Asai et al. (2020), *GRR + BERT-base* means the system whose retriever is GRR and reader is built on BERT-base. *: The results are reported by Asai et al. (2020).

Model / Dataset	EM	F1
BERT-Q	71.7	74.9
BERT-Q w. R/S	**77.2**	**80.4**
RoBERTa-Q	80.3	83.7
RoBERTa-Q w. R/S	**81.7**	**85.0**

Table 3: Results of our method and other baselines on the dev set of SQuAD2.0.

training method on the task requiring multi-hop evidence extraction and reasoning.

Besides, we also conducted experiments on the most common benchmark, SQuAD2.0. The results on development set shown in Table 3 have also verified the effectiveness of our proposed pre-training method.

5.3 Evaluation of Evidence Extraction

To evaluate the performance of our method for evidence extraction in the setting of implicit supervision (with only answers), we ranked sentences in a passage using their attention weights obtained in Equation 4 and chose those sentences with higher weights as the evidences.

As shown in Table 4, the models with our proposed pre-training tasks obtain considerable improvements on the precision and recall of evidence extraction, which verifies that our pre-training method is able to effectively equip PLMs with the capability for gathering evidence without explicit supervision. For a better illustration, we further provided two examples in Appendix C.

5.4 Effect of Different Masking Ratio During Pre-training

Table 5 shows the results of our model pre-trained with different masking ratios. Due to the small amount of entities contained in the document, we only consisdered the masking ratio of nouns as the variable. Formally, we considered three ratios: 30%, 60%, 90%, and an extra setting, where the entities and nouns are all kept and the RMLM task is also removed during pre-training.

Model	**P@1**	**R@1**	**P@2**	**R@2**
BERT-Q	21.83	9.66	20.24	17.73
BERT-Q w. R/S	**45.30**	**20.38**	**38.51**	**34.55**
RoBERTa-Q	28.25	12.45	26.93	23.74
RoBERTa-Q w. R/S	**35.34**	**15.76**	**30.33**	**26.85**

Table 4: Results of evidence extraction on the development set of Multi-RC.

Model/Dataset	RACE		Multi-RC		
	Dev	Test	Dev		
	Acc.	Acc.	EM	$F1_a$	$F1_m$
B.Q w.R/S (30%)	70.1	68.1	31.9	**73.8**	**76.3**
B.Q w.R/S (60%)	70.2	67.3	**32.0**	**73.8**	**76.3**
B.Q w.R/S (90%)	**70.4**	**68.2**	31.0	73.5	76.2
B.Q w.S (No Mask)	69.0	67.2	29.0	72.7	75.4

Table 5: Results on RACE and Multi-RC using models pre-trained with different mask ratios. *B.Q* means *BERT-Q*.

As shown in the table, with more possible clues being masked, the model tend to obtain better results on the downstream tasks. For example, BERT-Q w. R/S (90%) achieves the best accuracy on RACE, and BERT-Q w. R/S (60%) obtains the highest performance over Multi-RC. And all models that employ masking outperform BERT-Q w. S (no masking). The main reason can be that with more explicit information short-cut being eliminated, it is more difficult for models to collect potential clues, and PLMs are enhanced with stronger reasoning ability of evidence extraction. However, there also exists a trade-off: as higher masking ratio leads to more noise, it could worsen the mismatch between pre-training and fine-tuning, and cause performance degradation, e.g., BERT-Q w. R/S (90%) performs the worst on Multi-RC.

5.5 Performance in Low Resource Scenario

Figure 3 depicts the performance of BERT-Q w. R/S on the development and test set of RACE with limited training set. For each specific relative ratio, four reduced training sets are automatically generated using different random seeds and the corresponding accuracies are plotted on the figure. It is observed that with 70% training data, our model outperforms the baseline, BERT-Q, which was initialized using BERT and has not been further pre-

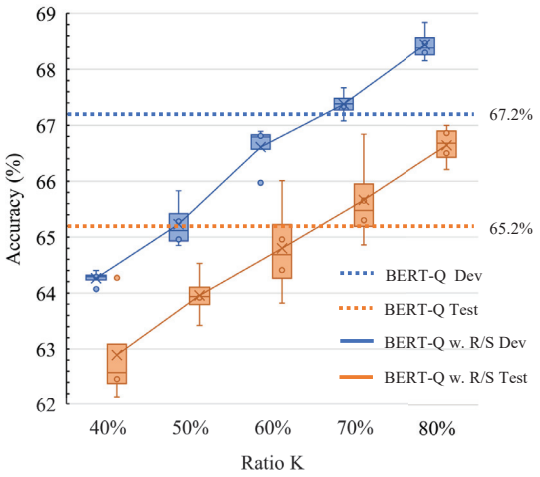


Figure 3: The accuracy of BERT-Q w. R/S on the development and test of RACE. The horizontal axis refers to the ratio K of training data compared to the original training set.

trained. The results indicate that our method can help to reduce the amount of annotated training data for downstream MRC tasks, which is especially useful in low resource scenarios.

6 Conclusion and Future Work

In this paper, we present a novel pre-training approach, REPT, to bridge the gap between pre-trained language models and machine reading comprehension through retrieval-based pre-training. Specifically, we design two retrieval-based pre-training tasks equipped with self-supervised learning, namely Surrounding Sentences Prediction (SSP) and Retreval based Masked Language Modeling (RMLM), to enhance PLMs with the capability of evidence extraction for MRC. The experiments over five different datasets validate the effectiveness of our proposed method. In the future, we plan to extend the proposed pre-training approach to the more challenging open-domain settings.

7 Acknowledgements

This work is supported by the National Key Research and Development Project of New Generation Artificial Intelligence, No.:2018AAA0102502, and the Alibaba Research Intern Program of Alibaba Group.

References

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2020. Learning to retrieve reasoning paths over wikipedia graph for question answering. In *ICLR*.

Lei Jimmy Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. 2016. Layer normalization. *CoRR*, abs/1607.06450.

Hyung Won Chung, Thibault Fevry, Henry Tsai, Melvin Johnson, and Sebastian Ruder. 2021. Rethinking embedding coupling in pre-trained language models. In *ICLR*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *NAACL*, pages 4171–4186. ACL.

Dheeru Dua, Yizhong Wang, Pradeep Dasigi, Gabriel Stanovsky, Sameer Singh, and Matt Gardner. 2019. DROP: A reading comprehension benchmark requiring discrete reasoning over paragraphs. In *NAACL*, pages 2368–2378. ACL.

Yuwei Fang, Siqi Sun, Zhe Gan, Rohit Pillai, Shuohang Wang, and Jingjing Liu. 2020. Hierarchical graph network for multi-hop question answering. In *EMNLP*, pages 8823–8838. ACL.

Yangyang Guo, Zhiyong Cheng, Liqiang Nie, Yibing Liu, Yinglong Wang, and Mohan Kankanhalli. 2019. Quantifying and alleviating the language prior problem in visual question answering. In *SIGIR*, pages 75–84.

Yangyang Guo, Liqiang Nie, Zhiyong Cheng, Feng Ji, Ji Zhang, and Alberto Del Bimbo. 2021. Adavqa: Overcoming language priors with adapted margin cosine loss. In *IJCAI*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. REALM: retrieval-augmented language model pre-training. *CoRR*, abs/2002.08909.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. DEBERTA: Decoding-enhanced bert with disentangled attention. In *ICLR*.

Dan Hendrycks and Kevin Gimpel. 2016. Bridging nonlinearities and stochastic regularizers with gaussian error linear units. *CoRR*, abs/1606.08415.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. 2020. Pretraining with contrastive sentence objectives improves discourse performance of language models. In *ACL*, pages 4859–4870.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S. Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *TACL*, 8:64–77.

Daniel Khashabi, Snigdha Chaturvedi, Michael Roth, Shyam Upadhyay, and Dan Roth. 2018. Looking Beyond the Surface: A challenge set for reading comprehension over multiple sentences. In *NAACL*, pages 252–262. ACL.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural Questions: a benchmark for question answering research. *TACL*, 7:452–466.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard H. Hovy. 2017. RACE: large-scale reading comprehension dataset from examinations. In *EMNLP*, pages 785–794. ACL.

Haejun Lee, Drew A. Hudson, Kangwook Lee, and Christopher D. Manning. 2020. SLM: learning a discourse language representation with sentence unshuffling. In *EMNLP*, pages 1551–1562. ACL.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *ACL*, pages 6086–6096.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. RoBERTa: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *ICLR*.

Yilin Niu, Fangkai Jiao, Mantong Zhou, Ting Yao, Jingfang Xu, and Minlie Huang. 2020. A self-training method for machine reading comprehension with soft evidence extraction. In *ACL*, pages 3916–3927.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018a. Know what you don't know: Unanswerable questions for squad. In *ACL*, pages 784–789.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018b. Know what you don't know: Unanswerable questions for squad. In *ACL*, pages 784–789.

Qiu Ran, Yankai Lin, Peng Li, Jie Zhou, and Zhiyuan Liu. 2019. Numnet: Machine reading comprehension with numerical reasoning. In *EMNLP-IJCNLP*, pages 2474–2484. ACL.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *TACL*, 7:249–266.

Kai Sun, Dian Yu, Jianshu Chen, Dong Yu, Yejin Choi, and Claire Cardie. 2019. DREAM: A challenge dataset and models for dialogue-based reading comprehension. *TACL*, 7:217–231.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE 2.0: A continual pre-training framework for language understanding. In *AAAI*, pages 8968–8975.

Alexandre Tamborrino, Nicola Pellicanò, Baptiste Pannier, Pascal Voitot, and Louise Naudin. 2020. Pre-training is (almost) all you need: An application to commonsense reasoning. In *ACL*, pages 3878–3887. ACL.

Alex Tamkin, Trisha Singh, Davide Giovanardi, and Noah D. Goodman. 2020. Investigating transferability in pretrained language models. In *EMNLP: Findings*, pages 1393–1401. ACL.

Yuval Varkel and Amir Globerson. 2020. Pre-training mention representations in coreference models. In *EMNLP*, pages 8534–8540. ACL.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *NeurIPS*, pages 5998–6008.

Shuohang Wang, Yuwei Fang, Siqi Sun, Zhe Gan, Yu Cheng, Jingjing Liu, and Jing Jiang. 2020. Cross-thought for sentence encoder pre-training. In *EMNLP*, pages 412–421. ACL.

Shuohang Wang, Mo Yu, Xiaoxiao Guo, Zhiguo Wang, Tim Klinger, Wei Zhang, Shiyu Chang, Gerry Tesauro, Bowen Zhou, and Jing Jiang. 2018. R^3: Reinforced ranker-reader for open-domain question answering. In *AAAI*, pages 5981–5988.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *CoRR*, abs/1910.03771.

An Yang, Quan Wang, Jing Liu, Kai Liu, Yajuan Lyu, Hua Wu, Qiaoqiao She, and Sujian Li. 2019a. Enhancing pre-trained language representations with rich knowledge for machine reading comprehension. In *ACL*, pages 2346–2357. ACL.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019b. Xlnet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*, pages 5754–5764.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *EMNLP*, pages 2369–2380. ACL.

Deming Ye, Yankai Lin, Jiaju Du, Zhenghao Liu, Peng Li, Maosong Sun, and Zhiyuan Liu. 2020. Coreferential reasoning learning for language representation. In *EMNLP*, pages 7170–7186. ACL.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. ReClor: A reading comprehension dataset requiring logical reasoning. In *ICLR*.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In *ACL*, pages 1441–1451. ACL.

Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul N. Bennett, and Saurabh Tiwary. 2020. Transformer-XH: Multi-evidence reasoning with extra hop attention. In *ICLR*.

A Implementation Detail

We built our model on Huggingface's Pytorch transformer repository (Wolf et al., 2019), and used AdamW (Loshchilov and Hutter, 2019) as the optimizer. We used the pre-trained BERT-base-uncased and RoBERTa-base checkpoint to initialize our encoder, and performed pre-training using 16 P100 GPUs simultaneously. The pre-training processes last around 16 hours for BERT and 4 days for RoBERTa, which takes 20,000 steps and 80,000 steps with the batch size as 512, respectively. All hyper-parameters can be found in Table 6 for pre-training and Table 7 for fine-tuning.

During constructing the training sample for pre-training, we controlled the masking ratio for entity and noun in query. For BERT, we masked 90% entities and 30% nouns. For RoBERTa, we constructed two datasets, where the masking ratios for entity and noun are set to 90%, 30% and 90%, 90%, respectively. And we mixed the two for jointly training. We also explored the effect of different masking ratios and the analysis is detailed in §5.

As for the fine-tuning stage, for multiple choice QA, we ran all experiments using for different random seeds (i.e., 33, 42, 57 and 67) and reported the average performance, except for ReClor, in which we only submitted the results obtained from the model which performs the best on development set to the leaderboard because the limitation of submission times. For Hotpot QA, we mainly followed the hyper-parameters of Asai et al. (2020) and thus did not repeat the experiments using different random seeds. Due to the submission limitation, we only submitted our best model on the development set to the leaderboard and reported its performance on test set.

B The Details About Modeling

B.1 Single-head Attention

To reduce the extra parameters introduced, we define a single-head attention mechanism compared to the multi-head one. Given the query matrix $\mathbf{Q}$, key matrix $\mathbf{K}$ and value matrix $\mathbf{V}$, the simple attention mechanism is formualted as:

$$\mathrm{Att}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \mathrm{softmax}((\mathbf{Q}\mathbf{W}+\mathbf{b})^\top \mathbf{K})\mathbf{V},$$

where $\mathbf{W}$ and $\mathbf{b}$ is the learnable parameters.

B.2 Normalized Feed-forward Network

We adopt a 2-layer feed-forward network with GeLU activation (Hendrycks and Gimpel, 2016) and layer normalization (Ba et al., 2016) to predict the masked entities and nouns. Following SpanBERT (Joshi et al., 2020), the Equation 7 is decomposed as:

$$\begin{cases} \mathbf{h}_0 = [\mathbf{h}_z; \mathbf{g}^q], \\ \mathbf{h}_1 = \mathrm{LayerNorm}(\mathrm{GeLU}(\mathbf{W}_3\mathbf{h}_0 + \mathbf{b}_3)), \\ \tilde{\mathbf{h}}_z^q = \mathrm{LayerNorm}(\mathrm{GeLU}(\mathbf{W}_4\mathbf{h}_1 + \mathbf{b}_4)). \end{cases}$$

C Case Study About Evidence Extraction

In §5.3, the results show that our pre-training method can augment the ability to extract the correct evidence. To give an intuitive clarification over this, we select two cases shown in Figure 4. As we can see, BERT-Q w. R/S and RoBERTa-Q w. R/S can select the correct evidence sentences, while the baselines models attend to the wrong sentences. Besides, Figure 5 shows the attention maps of the two groups of comparison. It can be observed that our pre-training approach can help the model learn a uniform attention distribution over the possible evidence sentences.

D Analysis of Extra Parameters Introduced

For fair comparison, we try to introduce as few additional parameters as possible. Since the output layer is highly task-specific and the single head-attention defined in Appendix B.1 is simple, we main analyze the extra parameters introduced for query representation learning defined in §3.3.1. A single layer of Transformer comprises of a multi-head attention module and a feed-forward network. As a result, the multi-head attention module generating the query representation has introduced 2.8% extra parameters compared with a 12-layer Transformer without consideration to the parameters in embedding layer and layer normalization.

HyperParam	BERT-base	RoBERTa-base
Peak Learning Rate	2e-4	5e-5
Learning Rate Decay	Linear	Linear
Batch Size	512	512
Max Steps	20,000	80,000
Warmup Steps	2,000	4,000
Weight Decay	0.01	0.01
Gradient Clipping	1.0	0.0
Adam ϵ	1e-6	1e-6
Adam β_1	0.9	0.9
Adam β_2	0.999	0.98
Max Sequence Length	512	512
Query Generator Dropout	0.1	0.1
SSP Dropout	0.1	0.1
RMLM Dropout	0.1	0.1
FP16 option level	O2	O2

Table 6: Hyper-parameters for pre-training.

HyperParam	RACE	DREAM	ReClor	MultiRC	Hotpot QA
Peak Learning Rate	4e-5♣/2e-5♠	3e-5♣/2e-5♠	2e-5♣/1e-5♠	3e-5	5e-5♣/3e-5♠
Learning Rate Decay	Linear	Linear	Linear	Linear	Linear
Batch Size	32♣/16♠	24	24	32	32♣/48♠
Epoch	4	8	10	8.0	3♣/4♠
Warmup Proportion	0.1♣/0.06♠	0.1	0.1	0.1	0.1
Weight Decay	0.01	0.01	0.01	0.01	0.01
Adam ϵ	1e-6	1e-6	1e-6	1e-6	1e-6♣/1e-8♠
Adam β_1	0.9	0.9	0.9	0.9	0.9
Adam β_2	0.999♣/0.98♠	0.999♣/0.98♠	0.999♣/0.98♠	0.999	0.999
Gradient Clipping	1.0♣/0.0♠	0.0♣/5.0♠	0.0	1.0	0.0
Max Sequence Length	512	512	256	512	384♣/386♠
Max Query Length	128	512	256	512	64
Dropout	0.1	0.1	0.1	0.1	0.1

Table 7: Hyper-parameters for fine-tuning. ♣: Hyper-parameters for BERT-based models. ♠: Hyper-parameters for RoBERTa-based models.

Case 1

Passage:

(0)A group of researchers at a remote jungle island outpost discover the natives are practicing voodoo and black magic. … (4)She returns years later as an adult with a group of mercenaries to attempt to uncover what happened to her parents. (5)Shortly after arriving at the island their boat's engine dies, stranding them. (6)Meanwhile elsewhere on the island a trio of hikers discover a cave, the same cave leading to the underground temple where the original curse was created. (7)After accidentally reviving the curse, the dead once again return to kill any who trespass on their island. (8)The mercenaries encounter their first zombie, who injures a member of the team. (9)Taking shelter in the remains of the old research facilities medical quarters they are soon joined by Chuck, the only surviving hiker. (10)Arming themselves with weapons left behind by the long dead research team, they make their stand as the dead once again rise. (11)One by one they are injured or killed, one of whom sacrifices himself to blow up the medical facility and his newly undead team members. (12)Jenny and Chuck flee, the only survivors remaining. (13)They stumble upon the cave once again, where the zombies appear and attack.

Question: Where did Chuck find weapons?

Option: From the previous research team.

Sentences Used: 9, 10.

BERT-Q: **Answer: False** **Evidence: 0**

BERT-Q w. R/S: **Answer: True** **Evidence: 10, 9**

Case 2

Passage:

(0)The film opens with Sunita, a medical student , and her friends working on a project about the human brain. (1)She wants to investigate the curious case of Sanjay Singhania, a notable city businessman, who is reported to have anterograde amnesia. (2)Her professor denies access to Sanjay's records as it is currently under criminal investigation. (3)Sunita, nonetheless, decides to investigate the matter herself. (4)Sanjay is introduced as he brutally murders a man. (5)He takes a Polaroid picture of the man, and writes on it ``done''. (6)It is revealed that Sanjay has anterograde amnesia where he loses his memory every 15 minutes. (7)Sanjay uses a system of photographs, notes, and tattoos on his body to recover his memory after each cycle. (8)It is revealed that Sanjay is ultimately out to avenge the death of his sweetheart Kalpana , and that he is systematically killing the people who were responsible for it. (9)His main target is ``Ghajini'', a notable social personality in the city. (10)Police Inspector Arjun Yadav, on the case of the serial murders, tracks Sanjay down to his flat and attacks and disables him. (11)Yadav finds two diaries where Sanjay has chronicled the events of 2005 and 2006. …

Question: Who denies Sunita access to Sanjay's records, who is reported to have anterograde amnesia, because they are under criminal investigation?

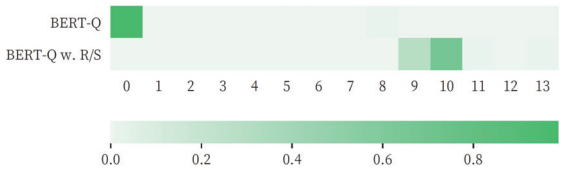
Option: Sunita's professor&Arjun Yadav.

Sentences Used: 1, 2.

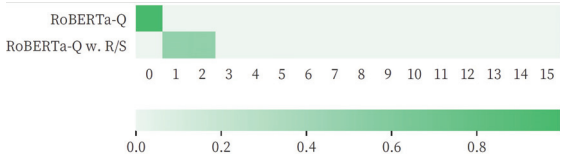
RoBERTa-Q: **Answer: False** **Evidence: 0**

RoBERTa-Q w. R/S: **Answer: True** **Evidence: 2, 1**

Figure 4: Two cases from the development set of Multi-RC.



(a) Normalized attention weights for Case 1 in Figure 4.



(b) Normalized attention weights for Case 2 in Figure 4.

Figure 5: Two cases of the normalized attention weights of evidence extraction.