To Answer or Not to Answer? Improving Machine Reading Comprehension Model with Span-based Contrastive Learning

Yunjie Ji, Liangyu Chen, Chenxiao Dou, Baochang Ma, Xiangang Li
Beike Inc., Beijing, China
{jiyunjie001,chenliangyu003,douchenxiao001,
mabaochang001,lixiangang002}@ke.com

Abstract

Machine Reading Comprehension with Unanswerable Questions is a difficult NLP task, challenged by the questions which can not be answered from passages. It is observed that subtle literal changes often make an answerable question unanswerable, however, most MRC models fail to recognize such changes. To address this problem, in this paper, we propose a span-based method of Contrastive Learning (spanCL) which explicitly contrast answerable questions with their answerable and unanswerable counterparts at the answer span level. With spanCL, MRC models are forced to perceive crucial semantic changes from slight literal differences. Experiments on SQuAD 2.0 dataset show that spanCL can improve baselines significantly, yielding 0.86~2.14 absolute EM improvements. Additional experiments also show that spanCL is an effective way to utilize generated questions.

1 Introduction

Machine Reading Comprehension (MRC) is an important task in Natural Language Understanding (NLU), aiming to answer specific questions through scanning a given passage(Hermann et al., 2015; Cui et al., 2016; Rajpurkar et al., 2018). As a fundamental NLU task, MRC also plays an essential role in many applications such as question answering and dialogue tasks (Chen et al., 2017; Gupta et al., 2020; Reddy et al., 2019). With the rapid development of pre-trained language models (PLMs), there is also a paradigm shift (Schick and Schütze, 2020; Dai et al., 2020; Sun et al., 2021) reformulating other NLP tasks (e.g. information extraction) into MRC format, especially for open-domain scenarios (Li et al., 2019; Yan et al., 2021a).

In most of the application scenarios, there exists a hypothesis that only answerable questions can be asked, which is somehow unrealistic and unreasonable. Thus, the model that is capable of distinguishing unanswerable questions is more welcomed than the model that can only give plausible answers (Rajpurkar et al., 2018). However, the challenge, that a slight literal change may transfer answerable questions into unanswerable ones, makes MRC models hard to gain such capability(Rajpurkar et al., 2018). For example, in Figure 1, the original answerable question becomes unanswerable by only replacing `Twilight` with `Australian`, but the small literal modification towards paraphrasing does not change the answer. Recent MRC models which predict answers using context-learning techniques and type-matching heuristics are not easy to perceive such subtle but crucial literal changes(Weissenborn et al., 2017; Jia and Liang, 2017). If different questions share many words in common, these models are most likely to give them the same answer, i.e., `2005` may be answered for all the three questions in Figure 1.

Passage:
The Legend of Zelda: Twilight Princess (Japanese: ゼルダの伝説 トワイライトプリンセス, Hepburn: Zeruda no Densetsu: Towairaito Purinsesu?) is an action-adventure game developed and published by Nintendo for the GameCube and Wii home video game consoles. It is the thirteenth installment in the The Legend of Zelda series. Originally planned for release on the GameCube in November ***2005***, Twilight Princess was delayed by Nintendo to allow its developers to refine the game, add more content, and port it to the Wii. The Wii version was released alongside the console in North America in November 2006, and in Japan, Europe, and Australia the following month. The GameCube version was released worldwide in December 2006.[b]

Original Question:
What year was the Legend of Zelda:***Twilight*** Princess originally planned for release?

Question Distortion:
What year was the Legend of Zelda: ***Australian*** Princess originally planned for release?

Question Paraphrase:
When was the legend of Zelda: ***Twilight*** Princess originally planned to be released?

Figure 1: Question Distortion and Question Paraphrase are derived by slightly changing Original Question.

To address the aforementioned challenge, we propose a span-based method of Contrastive Learning (spanCL) in this paper. By explicitly contrasting answerable questions with their paraphrases and their distortions, MRC models are forced to recognize the subtle but crucial literal changes. Using pre-trained language model (PLM) as encoder, most contrastive learning methods adopt `[CLS]` as

Findings of the Association for Computational Linguistics: NAACL 2022, pages 1292 - 1300
July 10-15, 2022 ©2022 Association for Computational Linguistics

the sentence representation (Luo et al., 2020; Wu et al., 2020; Gao et al., 2021; Yan et al., 2021b; Wang et al., 2021). However, in this problem, as the differences between contrastive questions are very subtle, [CLS] is inadequate to capture such small changes. To solve the challenge, we propose a novel learning method, which incorporates the comparative knowledge between answerable and unanswerable questions, and exploits the semantic information of answer spans to improve the sentence representation. Overall, our contributions are summarized as two folds:

- To improve MRC model's capability of distinguishing unanswerable questions, we propose a simple yet effective method called spanCL, which teaches the model to recognize crucial semantic changes from slight literal differences.
- Comprehensive experiments show that spanCL can yield substantial performance improvements of baselines. We also show that spanCL is an effective way to utilize generated questions.

2 Related Work

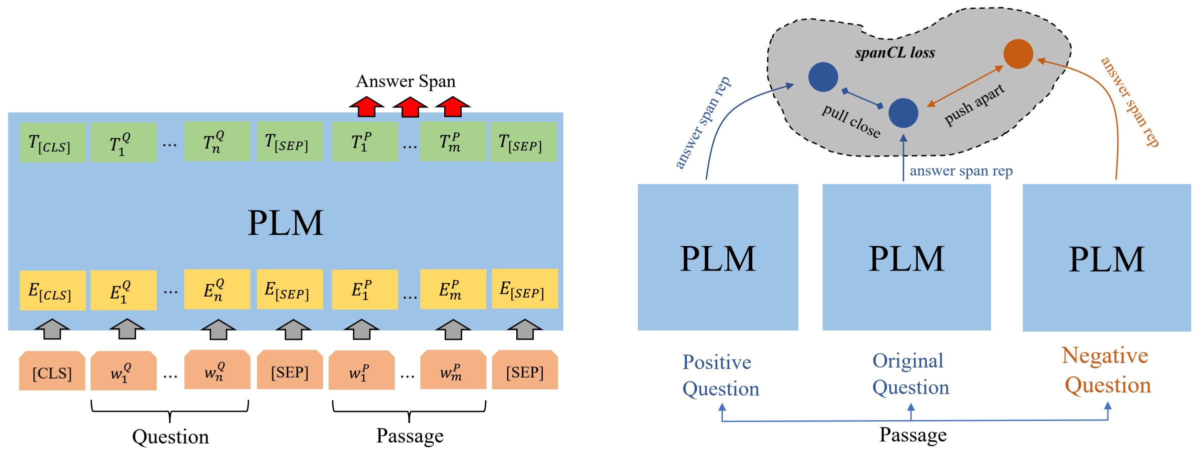
Models for MRC. With the help of various large-scale reading comprehension datasets (Hermann et al., 2015; Hill et al., 2015; Trischler et al., 2016; Rajpurkar et al., 2016; Lai et al., 2017; Rajpurkar et al., 2018), neural networks have achieved a great success on MRC in recent years. At first, these models are typically designed with a LSTM (Hochreiter and Schmidhuber, 1997) or CNN (LeCun et al., 1998) backbone, based on word embeddings (Mikolov et al., 2013; Pennington et al., 2014), leveraging various attention mechanisms to build interdependent representations of passage and question (Kadlec et al., 2016; Dhingra et al., 2016; Cui et al., 2016; Seo et al., 2016). Recently, pre-trained language models (PLMs) made a profound impact on NLP tasks (Radford et al., 2018; Devlin et al., 2018; Yang et al., 2019a; Liu et al., 2019; Lan et al., 2019; Clark et al., 2020; Brown et al., 2020; Fedus et al., 2021). With millions, billions even trillions of parameters, PLMs show a great capacity of capturing contextualized representations, and significantly boost the performance of MRC models.

MRC with Unanswerable Questions. Knowing what you do not know is a crucial aspect of model intelligence (Rajpurkar et al., 2018). In the field of MRC, a model should abstain from answering when no answer is available to the question. To deal with unanswerable questions, previous researchers mostly focused on designing a powerful answer verification module (Clark and Gardner, 2017; Liu et al., 2018; Kundu and Ng, 2018; Hu et al., 2019). Recently, a double checking strategy is proposed, in which an extra verifier is adopted to rectify the predicted answer (Hu et al., 2019; Back et al., 2019; Zhang et al., 2020a,b). Besides the idea of designing verification modules, some other studies try to solve the problem through data augmentation, namely to synthesize more QA pairs (Yang et al., 2019b; Alberti et al., 2019; Zhu et al., 2019b; Liu et al., 2020).

Contrastive Learning. To obtain rich representations of texts for down-stream NLP tasks, there have been numerous investigations of using contrastive objectives on strengthening supervised learning (Khosla et al., 2020; Gunel et al., 2020) and unsupervised learning (Chen et al., 2020; Gao et al., 2021) in various domains (He et al., 2020; Lin et al., 2020; Iter et al., 2020; Kipf et al., 2019). The main idea of contrastive learning (CL) is to learn textual representations by contrasting positive and negative examples, through concentrating the positives and alienating the negatives. In NLP tasks, CL is usually devoted to learning rich sentence representations (Luo et al., 2020; Wu et al., 2020), and the main difference between these methods is the approach to find positive and negative examples. Wang et al. (2021) argued that using hard negative examples in CL is helpful to improve the semantic robustness and sensitivity of pre-trained language models. Enlightened by the promising effects of CL, Kant et al. (2021) proposed to use CL in visual question answering. He focused on playing CL on MRC by comparing multiple answer candidates, but neglected the fact that not all questions can be answered through a given paragraph.

3 Approach

In this section, we first introduce the task of Machine Reading Comprehension with Unanswerable Questions (MRC-U). Then, a baseline MRC model based on PLM is described. At last, we propose a span-based contrastive learning method for MRC-U, named as spanCL. In this paper, question paraphrase and positive question, question distortion and negative question are used interchangeably.



(a) Base Model

(b) Span-based Contrastive Learning

Figure 2: (a) The baseline model for MRC-U. (b) Span-based contrastive learning on answer-span representation.

3.1 Task Description

In this paper, we focus on studying extractive MRC, in which the expected answer of a question is a word span of a given passage. Thus, given a textual question Q and a textual passage P, our goal is to find the answer span (y_s, y_e) to Q in P, where y_s is the answer start position in P and y_e is the answer end position in P.

3.2 Basic MRC Model

We use the model same as Devlin et al. (2018) as the basic model for MRC-U task. When a question and a passage are input, if the question is answerable, the model is expected to give a legal answer span (y_s, y_e) in the passage; if the question is unanswerable, the model is expected to output the `[CLS]` span $(0, 0)$, which indicates no related answer can be found in the passage. The overall structure of the network is presented in Figure 2.

For illustration, we denote the output of PLM's last layer as the sequence representation, $\boldsymbol{H} \in \mathbb{R}^{l\times d}$, where l is the sequence length and d is the dimension. Accordingly, the hidden representation of the i-th token in the sequence is denoted as $\boldsymbol{h}_i \in \boldsymbol{H}$. To find the start position of an answer, a start weight vector $\boldsymbol{w}_s \in \mathbb{R}^d$ is introduced to calculate the beginning possibility of each position. Formally, the probability that the answer starts at the i-th token is defined as

$$p_i^s = \frac{\exp(\boldsymbol{h}_i \cdot \boldsymbol{w}_s)}{\sum\limits_{j\leq l} \exp(\boldsymbol{h}_j \cdot \boldsymbol{w}_s)}. \tag{1}$$

Similarly, with an end weight vector $\boldsymbol{w}_e \in \mathbb{R}^d$, the probability that the answer ends at the i-th token is defined as

$$p_i^e = \frac{\exp(\boldsymbol{h}_i \cdot \boldsymbol{w}_e)}{\sum\limits_{j\leq l} \exp(\boldsymbol{h}_j \cdot \boldsymbol{w}_e)}. \tag{2}$$

For learning, the cross-entropy loss on identifying the answer start and end positions is taken as the training objective as

$$\mathcal{L}_{span} = -\log(p^s_{y_s}) - \log(p^e_{y_e}), \tag{3}$$

where y_s and y_e are the start and end positions of the true answer span. With the learnt model, the output answer span (y'_s, y'_e) is predicted according to

$$(y'_s, y'_e) = \arg\max_{(i,j)|i\leq j} \boldsymbol{h}_i \cdot \boldsymbol{w}_s + \boldsymbol{h}_j \cdot \boldsymbol{w}_e. \tag{4}$$

3.3 Span-based Contrastive Learning

In this section, spanCL is introduced from two aspects. First, considering the contrastive idea of CL, we give the details about how the positive and negative examples are generated. Second, the training objective of spanCL is presented.

Positive Examples. In our method, we define the positive examples as the questions which have slight literal differences but the same answers with their original questions. Back Translation is an effective data augmentation method (Xie et al., 2019; Zhang et al., 2017; Zhu et al., 2019a), in which a text is first translated to a target language (e.g. France) from its source language (e.g. English), and then back translated to the source language. The final back-translated text is taken as the example of augmentation. Thanks to Back Translation, the produced examples are lexically different

Strategy	Example
Negation	Original question: What was Beyonce's role in Destiny's Child? Negative question: What wasn't Beyonce's role in Destiny's Child?
Entity replacement	Original question: What native people lived in the San Diego area before the Europeans arrived? Negative question: What native people lived in the San Diego area before the Mexicans arrived?
Antonym	Original question: What part of Gothic buildings are often found terminated with enormous pinnacles? Negative question: What other part of Gothic buildings are often found terminated with small pinnacles?

Table 1: Strategies used to generate negative questions.

but semantically same with the original example. Specifically, for each question, we first produce three question paraphrases by Back Translation using three languages. Then we select the question that has the most literal differences with the original question as the positive question.

Negative Examples. In our method, we define the negative examples as the questions which have slight literal differences and not the same answers with their original questions. Three simple strategies are adopted to produce negative examples as the following.

- **Negation**. A negation word is inserted or removed from the original question.
- **Antonym**. First, spaCy [1] is utilized to conduct segmentation and POS for the original question. Then, one of the words (verbs, nouns, adjectives, or adverbs) are randomly replaced with its antonym.
- **Entity Replacement**. With an answerable question, one of its entity words is randomly placed with another entity word, which has the same entity type but does not appear in any questions.

Table 1 shows several negative examples derived by these strategies. Note that question generation is not the main topic of this paper.

Span-based Contrastive Learning. Using PLM as the encoder, `[CLS]` usually serve as the sentence representation in CL (Gao et al., 2021; Wang et al., 2021; Yan et al., 2021b). When the difference between the original question and its paraphrase or distortion is very subtle, a single `[CLS]` token is not adequate to capture the difference, making the model hard to answer such question. Therefore, we propose to improve MRC models by contrasting these questions according to their answer-span representations. Specifically, given a question Q_{org} and its answer span (y_s, y_e), through the augmentation methods mentioned previously, we synthesize one positive question Q_{pos} and one negative question Q_{neg}. Based on the definition of positive examples and negative examples, (y_s, y_e) is the answer span to both Q_{pos} and Q_{org} but not to Q_{neg}. Denote $\boldsymbol{h}_{y_s}^{Q_{org}}$ and $\boldsymbol{h}_{y_e}^{Q_{org}}$ as the representation vectors of the y_s-th token and y_e-th token in the input passage P for the question Q_{org}, $\boldsymbol{h}_{y_s}^{Q_{pos}}$ and $\boldsymbol{h}_{y_e}^{Q_{pos}}$ as those for Q_{pos}, and $\boldsymbol{h}_{y_s}^{Q_{neg}}$ and $\boldsymbol{h}_{y_e}^{Q_{neg}}$ as those for Q_{neg}. The concatenation of $h_{y_s}^{Q_{org}}$ and $\boldsymbol{h}_{y_e}^{Q_{org}}$ is used as the answer-span representation to Q_{org} and denoted as $\boldsymbol{z}^{Q_{org}}$. Similarly, the answer-span representation to Q_{pos} and Q_{neg} are denoted as $\boldsymbol{z}^{Q_{pos}}$ and $\boldsymbol{z}^{Q_{neg}}$ respectively. Then, our span-based contrastive loss is calculated as

$$\mathcal{L}_{spanCL} = -\log \frac{\exp(\Phi(\boldsymbol{z}^{Q_{org}}, \boldsymbol{z}^{Q_{pos}})/\tau)}{\exp(\Phi(\boldsymbol{z}^{Q_{org}}, \boldsymbol{z}^{Q_{pos}})/\tau)) + \exp(\Phi(\boldsymbol{z}^{Q_{org}}, \boldsymbol{z}^{Q_{neg}})/\tau))} \quad (5)$$

where $\Phi(\boldsymbol{u}, \boldsymbol{v}) = \boldsymbol{u}^\top \boldsymbol{v} / \|\boldsymbol{u}\| \|\boldsymbol{v}\|$ computes similarity between $\boldsymbol{u}$ and $\boldsymbol{v}$ and $\tau > 0$ is a scalar temperature parameter. With the definition, the final objective loss of our method is presented as the following:

$$\mathcal{L} = \lambda_1 \mathcal{L}_{span} + \lambda_2 \mathcal{L}_{spanCL}. \quad (6)$$

4 Experiments

4.1 Datasets and Metrics

We evaluate our method on the well-known dataset SQuAD 2.0 (Rajpurkar et al., 2018), which covers the questions of SQuAD1.1 (Rajpurkar et al., 2016) with new unanswerable questions written adversarially by crowdworkers to imitate the answerable ones. Moreover, for each unanswerable question, a

[1] https://github.com/explosion/spaCy

plausible answer span is annotated, which indicates the incorrect answer obtained by type-matching heuristics. The training dataset contains 87k answerable and 43k unanswerable questions, and half of the examples in the development set are unanswerable.

Two official metrics are used to evaluate the model performance on SQuAD 2.0: Exact Match (EM) and F1. EM is used to compute the percentage of predictions that match ground truth answers exactly. F1 is a softer metric, which measures the average overlap between the prediction and ground truth answer at token level.

4.2 Experimental Setup

MRC Model. We adopt the model introduced in 3.2 with various PLM encoders for the MRC-U task. Bert (Devlin et al., 2018), RoBERTa (Liu et al., 2019), ALBERT (Lan et al., 2019) are selected in our experiments. We download the pre-trained weights from Hugging Face[2].

Training Data Construction. For each original answerable question, we use Back Translation to generate its paraphrase. In SQuAD 2.0, we can find the negative questions for 18,541 answerable questions in the original dataset. For the rest 68,280 answerable questions, we use our augmentation strategies to generate negative questions.

During our training, the span loss is calculated based on Q_{org} and Q_{neg}. In section 4.3, we will explain why Q_{pos} is discarded for calculating span loss.

Hyper-parameters. We use the default hyper-parameter settings for the SQuAD 2.0 task. Specifically, we set maximum sequence length, doc stride, maximum query length and maximum answer length to 512, 128, 64 and 30. For fine-tuning our model, we set the learning rate, batch size, training epoch and warm-up rate to 2e-5, 12, 2 and 0.1. The temperature in spanCL is set to 0.05. The weights of span loss and spanCL loss are $\lambda_1 = \lambda_2 = 0.5$. For each time, we fix the random seed, ensuring our results are reproducible. We run our experiments on two Tesla A100 40G GPUs with 5 GPU hours to train a model.

4.3 Main Results

From Table 2, we notice that spanCL improves the performance of each baseline model, yielding 0.86~2.14 absolute EM improvement and 0.76~2.0 absolute F1 improvement, demonstrating spanCL is model-agnostic and effective.

[2]https://huggingface.co/bert-base-uncased

Model	Dev		Δ	
	EM	F1	EM	F1
BERT_{base}	73.37	76.34	-	-
+ spanCL	75.51	78.34	+2.14	+2.00
BERT_{large}	78.88	81.85	-	-
+ spanCL	79.76	82.61	+0.88	+0.76
RoBERTa_{base}	78.85	81.42	-	-
+ spanCL	80.18	82.84	+1.33	+1.42
RoBERTa_{large}	86.12	88.88	-	-
+ spanCL	86.98	89.70	+0.86	+0.82
ALBERT_{base}	77.84	81.27	-	-
+ spanCL	79.52	82.97	+1.68	+1.7
ALBERT_{large}	79.99	83.27	-	-
+ spanCL	81.51	84.67	+1.52	+1.4

Table 2: Results (%) on the dev set of SQuAD 2.0.

Model	EM		Dev	
	HasAns	NoAns	EM	F1
BERT_{base}	70.31	74.76	73.37	76.34
+ pos	**72.57**	68.93	72.26	75.22
+ neg	67.05	**87.74**	74.02	76.38
+ pos&neg	66.16	78.48	72.59	75.37
+ spanCL	72.52	75.91	**75.51**	**78.34**

Table 3: Training with spanCL vs Training with expanded datasets.

As additional training data (i.e. the extra positive and negative questions) is used, it is necessary to analyze if the improvements are merely brought by this additional data. We conduct experiments by training with different datasets and display the results in Table 3. BERT_{base} means training BERT_{base} with original SQuAD 2.0 training set. "+pos" and "+neg" mean expanding the original training set with generated positive questions and generated negative questions respectively. Surprisingly, Simply expanding the training set can not guarantee the performance improvement. We find that adding positive examples into the training set does not improve the performance of MRC model. One possible reason is that the positive questions make the model over insensitive and ignore slight literal changes, which is inappropriate for MRC-U task. By comparing BERT_{base} with "+neg", we find that training with more negative examples, the model tends to predict more `NoAns` and achieve a high performance on `NoAns`, while the performance on the `HasAns` drops a lot and the overall improvement of EM is much less than "+spanCL". From the results in Table 3, we can conclude that spanCL is effective to utilize the generated questions.

Model	EM	F1
BERT$_{base}$	73.37	76.34
+ CRQDA (Liu et al., 2020)	75.80	78.70
+ spanCL with simple negatives	75.51	78.34
+ spanCL with CRQDA	**76.12**	**79.09**

Table 4: Performance of spanCL with different synthetic negative questions.

Model	EM	F1
BERT$_{base}$	73.37	76.34
+ spanCL	**75.51**	**78.34**
+ CL with [CLS] reps	74.18	77.05
+ CL with span and [CLS] reps	73.82	76.86

Table 5: Results (%) with different question representations used in the contrastive learning.

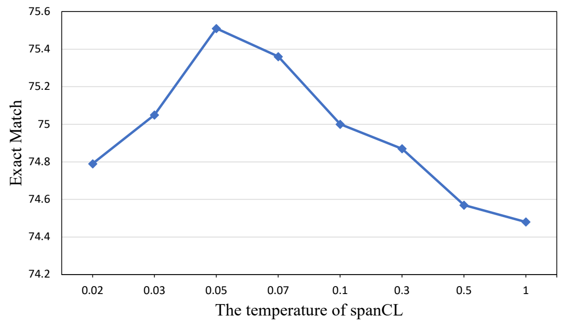


Figure 3: The influence of different temperatures in spanCL. The best performance is achieved when the temperature is set to 0.05. BERT$_{base}$ is adopted as the base model.

Base Model	Training schemes	EM	F1
BERT$_{base}$	Joint	**75.51**	**78.34**
BERT$_{base}$	Alternate	74.67	77.08
BERT$_{base}$	pre-train then finetune	72.19	74.87
BERT$_{large}$	Joint	**79.76**	**82.61**
BERT$_{large}$	Alternate	79.50	82.55
BERT$_{large}$	pre-train then finetune	77.77	80.30

Table 6: Results (%) with different training schemes.

4.4 Influence of Negative Examples

The unanswerable questions generated by our strategies are rather plain. We believe spanCL can further boost the performance by high-quality unanswerable questions. Liu et al. (2020) proposed a context-relevant generation method called CRQDA, which generates delicate negative questions [3]. In table 4, "+CRQDA" denotes training the baseline model with the dataset including the delicate negative questions generated by CRQDA. "+spanCL with simple negatives" denotes applying spanCL with negative questions generated by our three strategies. "+spanCL with CRQDA" denotes applying spanCL with negative questions generated by CRQDA. Comparing "+spanCL with simple negatives" with "+spanCL with CRQDA", we find that spanCL can further boost the performance by delicate negative questions.

4.5 Influence of Temperature

The temperature τ in spanCL loss (Equation 5) is used to control the smoothness of the distribution normalized by the softmax operation. A large temperature smoothes the distribution while a small temperature sharpens the distribution. As shown in the Figure 3, spanCL is sensitive to the temperature value. In general, small temperature results in better performance. A practical temperature can be obtained within a small range (from about 0.02 to 0.1). We select 0.05 as the temperature in our experiments.

[3]https://github.com/dayihengliu/CRQDA

4.6 Selection of Question Representations

In this paper, we argue that the answer span representation is better than `[CLS]`. We conduct experiments with different question representations in this section. When applying CL with `[CLS]` representations, we add a classification layer on the top of `[CLS]` to determine if a question is answerable or not (Zhang et al., 2020b), making the representation of `[CLS]` acquire the information of the question's answerability. We also play CL with both `[CLS]` and answer-span representations, in which two CL losses are calculated together. From Table 5, we can see that `CL with [CLS] reps` improves the model performance but the improvement is small than that from spanCL, and the combination of the two CL losses can confuse the model and result in a little improvement.

4.7 Comparison between Different Training Schemes

There are three training schemes to combine the span loss and spanCL loss: 1) joint training, in which these two losses are used together in each training step; 2) alternate training, in which the model is updated with spanCL loss after every M updates with span loss; 3) pre-train and fine-tune, in which we first pre-train the model with spanCL loss and then fine-tune it with span loss. For alternate training, we select M from $\{1, 2, 3\}$ and find $M = 2$ gives the best results. From Table

Passage: The Normans (Norman: Nourmands; French: Normands; Latin: Normanni) were the people who in the 10th and 11th centuries gave their name to ***Normandy***, a region in France. They were descended from Norse (\"Norman\" comes from \"Norseman\") raiders and pirates from Denmark, Iceland and Norway who, under their leader Rollo, agreed to swear fealty to King Charles III of West Francia.

....

The Norman dynasty had a major ***political, cultural and military*** impact on medieval Europe and even the Near East. The Normans were famed for their martial spirit and eventually for their Christian piety, becoming exponents of the Catholic orthodoxy into which they assimilated. They adopted the Gallo-Romance language of the Frankish land they settled, their dialect becoming known as Norman, Normaund or Norman French, an important literary language.

Question: What is France a region of ?

Answer from baseline: Normandy ❌

Answer from baseline+spanCL: no answer ✅

Question: What type of major impact did the Norman dynasty have on modern Europe ?

Answer from baseline: political, cultural and military ❌

Answer from baseline+spanCL: no answer ✅

Figure 4: Qualitative Examples.

6, we conclude that joint training gives the best performance and alternate training performs a little worse. Surprisingly, with the pre-train and fine-tune training scheme, the model performs worse than the baseline model. We guess this is because without the supervision of answer-span knowledge, it is hard to learn useful question representations.

4.8 Qualitative Analysis

We qualitatively analyze two representative unanswerable questions in Figure 4. It can be seen that the baseline model predicts a plausible answer for each question while the baseline model trained with spanCL abstain from answering.

To correctly answer the first question, the model is asked to learn the question's semantics in sentence level. To correctly answer the second question, the model is asked to recognize the literal change in word level. SpanCL can help the model perceive such crucial differences between the question and passage from both semantic and lexical aspects, and thus enable the baseline model to abstain from answering for these two questions.

5 Conclusion

In this paper, we propose a span-based method of Contrastive Learning (spanCL) to solve the MRC task with Unanswerable Questions. SpanCL is devised based on the fact that an answerable question can become unanswerable with slight literal changes. By explicitly contrasting an answerable question with its paraphrase and distortion at the answer span level, MRC models can be taught to perceive subtle but crucial literal changes. Experimental results demonstrate that spanCL is model-agnostic and can improve MRC models significantly. Additional experiments show that spanCL is more effective to utilize the generated questions than other methods. In addition, it should be noticed that how to generate high-quality question examples is not fully investigated in this paper, which may introduce a performance bottleneck to spanCL. Therefore, a study on question generation compatible with spanCL is encouraged in the future.

References

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic qa corpora generation with roundtrip consistency. *arXiv preprint arXiv:1906.05416*.

Seohyun Back, Sai Chetan Chinthakindi, Akhil Kedia, Haejun Lee, and Jaegul Choo. 2019. Neurquri: Neural question requirement inspector for answerability prediction in machine reading comprehension. In *International Conference on Learning Representations*.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. 2020. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR.

Christopher Clark and Matt Gardner. 2017. Simple and effective multi-paragraph reading comprehension. *arXiv preprint arXiv:1710.10723*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Yiming Cui, Zhipeng Chen, Si Wei, Shijin Wang, Ting Liu, and Guoping Hu. 2016. Attention-over-attention neural networks for reading comprehension. *arXiv preprint arXiv:1607.04423*.

Xiang Dai, Sarvnaz Karimi, Ben Hachey, and Cecile Paris. 2020. An effective transition-based model for discontinuous ner. *arXiv preprint arXiv:2004.13454*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Bhuwan Dhingra, Hanxiao Liu, Zhilin Yang, William W Cohen, and Ruslan Salakhutdinov. 2016. Gated-attention readers for text comprehension. *arXiv preprint arXiv:1606.01549*.

William Fedus, Barret Zoph, and Noam Shazeer. 2021. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *arXiv preprint arXiv:2101.03961*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Beliz Gunel, Jingfei Du, Alexis Conneau, and Ves Stoyanov. 2020. Supervised contrastive learning for pre-trained language model fine-tuning. *arXiv preprint arXiv:2011.01403*.

Somil Gupta, Bhanu Pratap Singh Rawat, and Hong Yu. 2020. Conversational machine comprehension: a literature review. *arXiv preprint arXiv:2006.00671*.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9729–9738.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. *Advances in neural information processing systems*, 28:1693–1701.

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

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.

Minghao Hu, Furu Wei, Yuxing Peng, Zhen Huang, Nan Yang, and Dongsheng Li. 2019. Read+ verify: Machine reading comprehension with unanswerable questions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6529–6537.

Dan Iter, Kelvin Guu, Larry Lansing, and Dan Jurafsky. 2020. Pretraining with contrastive sentence objectives improves discourse performance of language models. *arXiv preprint arXiv:2005.10389*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. *arXiv preprint arXiv:1707.07328*.

Rudolf Kadlec, Martin Schmid, Ondrej Bajgar, and Jan Kleindienst. 2016. Text understanding with the attention sum reader network. *arXiv preprint arXiv:1603.01547*.

Yash Kant, Abhinav Moudgil, Dhruv Batra, Devi Parikh, and Harsh Agrawal. 2021. Contrast and classify: Training robust vqa models.

Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. 2020. Supervised contrastive learning. *arXiv preprint arXiv:2004.11362*.

Thomas Kipf, Elise van der Pol, and Max Welling. 2019. Contrastive learning of structured world models. *arXiv preprint arXiv:1911.12247*.

Souvik Kundu and Hwee Tou Ng. 2018. A nil-aware answer extraction framework for question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4243–4252.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. Race: Large-scale reading comprehension dataset from examinations. *arXiv preprint arXiv:1704.04683*.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. 1998. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2019. A unified mrc framework for named entity recognition. *arXiv preprint arXiv:1910.11476*.

Zibo Lin, Deng Cai, Yan Wang, Xiaojiang Liu, Hai-Tao Zheng, and Shuming Shi. 2020. The world is not binary: Learning to rank with grayscale data for dialogue response selection. *arXiv preprint arXiv:2004.02421*.

Dayiheng Liu, Yeyun Gong, Jie Fu, Yu Yan, Jiusheng Chen, Jiancheng Lv, Nan Duan, and Ming Zhou. 2020. Tell me how to ask again: Question data augmentation with controllable rewriting in continuous space. *arXiv preprint arXiv:2010.01475*.

Xiaodong Liu, Wei Li, Yuwei Fang, Aerin Kim, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for squad 2.0. *arXiv preprint arXiv:1809.09194*.

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. *arXiv preprint arXiv:1907.11692*.

Fuli Luo, Pengcheng Yang, Shicheng Li, Xuancheng Ren, and Xu Sun. 2020. Capt: Contrastive pre-training for learning denoised sequence representations. *arXiv preprint arXiv:2010.06351*.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Siva Reddy, Danqi Chen, and Christopher D Manning. 2019. Coqa: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Timo Schick and Hinrich Schütze. 2020. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

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

Tianxiang Sun, Xiangyang Liu, Xipeng Qiu, and Xuanjing Huang. 2021. Paradigm shift in natural language processing. *arXiv preprint arXiv:2109.12575*.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2016. Newsqa: A machine comprehension dataset. *arXiv preprint arXiv:1611.09830*.

Dong Wang, Ning Ding, Piji Li, and Hai-Tao Zheng. 2021. Cline: Contrastive learning with semantic negative examples for natural language understanding. *arXiv preprint arXiv:2107.00440*.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural qa as simple as possible but not simpler. *arXiv preprint arXiv:1703.04816*.

Zhuofeng Wu, Sinong Wang, Jiatao Gu, Madian Khabsa, Fei Sun, and Hao Ma. 2020. Clear: Contrastive learning for sentence representation.

Qizhe Xie, Zihang Dai, Eduard Hovy, Minh-Thang Luong, and Quoc V Le. 2019. Unsupervised data augmentation for consistency training. *arXiv preprint arXiv:1904.12848*.

Hang Yan, Tao Gui, Junqi Dai, Qipeng Guo, Zheng Zhang, and Xipeng Qiu. 2021a. A unified generative framework for various ner subtasks. *arXiv preprint arXiv:2106.01223*.

Yuanmeng Yan, Rumei Li, Sirui Wang, Fuzheng Zhang, Wei Wu, and Weiran Xu. 2021b. Consert: A contrastive framework for self-supervised sentence representation transfer. *arXiv preprint arXiv:2105.11741*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019a. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Ziqing Yang, Yiming Cui, Wanxiang Che, Ting Liu, Shijin Wang, and Guoping Hu. 2019b. Improving machine reading comprehension via adversarial training. *arXiv preprint arXiv:1911.03614*.

Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. 2017. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*.

Zhuosheng Zhang, Yuwei Wu, Junru Zhou, Sufeng Duan, Hai Zhao, and Rui Wang. 2020a. Sg-net: Syntax-guided machine reading comprehension. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9636–9643.

Zhuosheng Zhang, Junjie Yang, and Hai Zhao. 2020b. Retrospective reader for machine reading comprehension. *arXiv preprint arXiv:2001.09694*.

Chen Zhu, Yu Cheng, Zhe Gan, Siqi Sun, Tom Goldstein, and Jingjing Liu. 2019a. Freelb: Enhanced adversarial training for natural language understanding. *arXiv preprint arXiv:1909.11764*.

Haichao Zhu, Li Dong, Furu Wei, Wenhui Wang, Bing Qin, and Ting Liu. 2019b. Learning to ask unanswerable questions for machine reading comprehension. *arXiv preprint arXiv:1906.06045*.