Learning to Ask Unanswerable Questions for Machine Reading Comprehension

Haichao Zhu[†][*], **Li Dong**[‡], **Furu Wei**[‡], **Wenhui Wang**[‡], **Bing Qin**[†♮], **Ting Liu**[†♮]
[†]Harbin Institute of Technology, Harbin, China
[‡]Microsoft Research, Beijing, China
[♮]Peng Cheng Laboratory, Shenzhen, China
{hczhu,qinb,tliu}@ir.hit.edu.cn
{lidong1,fuwei,wenwan}@microsoft.com

Abstract

Machine reading comprehension with unanswerable questions is a challenging task. In this work, we propose a data augmentation technique by automatically generating relevant unanswerable questions according to an answerable question paired with its corresponding paragraph that contains the answer. We introduce a pair-to-sequence model for unanswerable question generation, which effectively captures the interactions between the question and the paragraph. We also present a way to construct training data for our question generation models by leveraging the existing reading comprehension dataset. Experimental results show that the pair-to-sequence model performs consistently better compared with the sequence-to-sequence baseline. We further use the automatically generated unanswerable questions as a means of data augmentation on the SQuAD 2.0 dataset, yielding 1.9 absolute F1 improvement with BERT-base model and 1.7 absolute F1 improvement with BERT-large model.

本文提出了一种对
MRC的数据增强
技术。通过可回答
的问题及包含答案
的passage 生成
相关的不可回答的问题

本文也提出了一种
方法，利用现有的MRC
数据集构造问题生成
模型所需的训练数据.

使用生成的不可回答问题
作为数据增强.

1 Introduction

Extractive reading comprehension (Hermann et al., 2015; Rajpurkar et al., 2016) obtains great attentions from both research and industry in recent years. End-to-end neural models (Seo et al., 2017; Wang et al., 2017; Yu et al., 2018) have achieved remarkable performance on the task if answers are assumed to be in the given paragraph. Nonetheless, the current systems are still not good at deciding whether no answer is presented in the context (Rajpurkar et al., 2018). For unanswerable questions, the systems are supposed to abstain from answering rather than making unreliable guesses, which is an embodiment of language understanding ability.

放弃

目前的MRC
模型对于判
断问题是否
可回答仍然
做不好.

Title: Victoria (Australia)
Paragraph: ...Public schools, also known as state or government schools, are funded and run directly by the Victoria Department of Education . Students do not pay tuition fees, but some extra costs are levied. Private fee-paying schools include parish schools ...

Ans. Question: What organization runs *the public schools* in Victoria?
UnAns. Question: What organization runs *the waste management* in Victoria?

(Plausible) Answer: Victoria Department of Education

Figure 1: An example taken from the SQuAD 2.0 dataset. The annotated (plausible) answer span in the paragraph is used as a pivot to align the pair of answerable and unanswerable questions.

We attack the problem by automatically generating unanswerable questions for data augmentation to improve question answering models. The generated unanswerable questions should not be too easy for the question answering model so that data augmentation can better help the model. For example, a simple baseline method is randomly choosing a question asked for another paragraph, and using it as an unanswerable question. However, it would be trivial to determine whether the retrieved question is answerable by using word-overlap heuristics, because the question is irrelevant to the context (Yih et al., 2013). In this work, we propose to generate unanswerable questions by editing an answerable question and conditioning on the corresponding paragraph that contains the answer. So the generated unanswerable questions are more lexically similar and relevant to the context. Moreover, by using the answerable question as a prototype and its answer span as a plausible answer, the generated examples can provide more discriminative training signal to the question answering model.

避免trivial
的不可回答
问题

[*] Contribution during internship at Microsoft Research Asia.

构造训练数据集

To create training data for unanswerable question generation, we use (plausible) answer spans in paragraphs as pivots 支点 to align pairs of answerable questions and unanswerable questions. As shown in Figure 1, the answerable and unanswerable questions of a paragraph are aligned through the text span "*Victoria Department of Education*" for being both the answer and plausible answer. These two questions are lexically similar and both asked with the same answer type in mind. In this way, we obtain the data with which the models can learn to ask unanswerable questions by editing answerable ones with word exchanges, negations, etc. Consequently, we can generate a mass of unanswerable questions with existing large-scale machine reading comprehension datasets.

Inspired by the neural reading comprehension models (Xiong et al., 2017; Huang et al., 2018), we introduce a pair-to-sequence model to better capture the interactions between questions and paragraphs. The proposed model first encodes input question and paragraph separately, and then conducts attention-based matching to make them aware of each other. Finally, the context-aware representations are used to generate outputs. To facilitate the use of context words during the generation process, we also incorporate the copy mechanism (Gu et al., 2016; See et al., 2017).

Experimental results on the unanswerable question generation task shows that the pair-to-sequence model generates consistently better results over the sequence-to-sequence baseline and performs better with long paragraphs than with short answer sentences. Further experimental results show that the generated unanswerable questions can improve multiple machine reading comprehension models. Even using BERT fine-tuning as a strong reading comprehension model, we can still obtain a 1.9% absolute improvement of F1 score with BERT-base model and 1.7% absolute F1 improvement with BERT-large model.

2 Related Work

Machine Reading Comprehension (MRC) Various large-scale datasets (Hermann et al., 2015; Rajpurkar et al., 2016; Nguyen et al., 2016; Joshi et al., 2017; Rajpurkar et al., 2018; Kocisky et al., 2018) have spurred rapid progress on machine reading comprehension in recent years. SQuAD (Rajpurkar et al., 2016) is an extractive benchmark whose questions and answers spans are annotated by humans. Neural reading comprehension systems (Wang and Jiang, 2017; Seo et al., 2017; Wang et al., 2017; Hu et al., 2018; Huang et al., 2018; Liu et al., 2018; Yu et al., 2018; Wang et al., 2018) have outperformed humans on this task in terms of automatic metrics. The SQuAD 2.0 dataset (Rajpurkar et al., 2018) extends SQuAD with more than $50,000$ crowdsourced unanswerable questions. So far, neural reading comprehension models still fall behind humans on SQuAD 2.0. Abstaining from answering when no answer can be inferred from the given document does require more understanding than barely extracting an answer.

Question Generation for MRC In recent years, there has been an increasing interest in generating questions for reading comprehension. Du et al. (2017) show that neural models based on the encoder-decoder framework can generate significantly better questions than rule-based systems (Heilman and Smith, 2010). To generate answer-focused questions, one can simply indicate the answer positions in the context with extra features (Yuan et al., 2017; Zhou et al., 2018; Du and Cardie, 2018; Sun et al., 2018; Dong et al., 2019). Song et al. (2018) and Kim et al. (2019) separate answer representations for further matching. Yao et al. (2018) introduce a latent variable for capturing variability and an observed variable for controlling question types. In summary, the above mentioned systems aim to generate answerable questions with certain context. On the contrary, our goal is to generate unanswerable questions.

Adversarial Examples for MRC To evaluate the language understanding ability of pre-trained systems, Jia and Liang (2017) construct adversarial examples by adding distractor sentences that do not contradict question answering for humans to the paragraph. Clark and Gardner (2018) and Tan et al. (2018) use questions to retrieve paragraphs that do not contain the answer as adversarial examples. Rajpurkar et al. (2018) create unanswerable questions through rigid rules, which swap entities, numbers and antonyms of answerable questions. It has been shown that adversarial examples generated by rule-based systems are much easier to detect than ones in the SQuAD 2.0 dataset.

Data Augmentation for MRC Several attempts have been made to augment training data for machine reading comprehension. We catego-

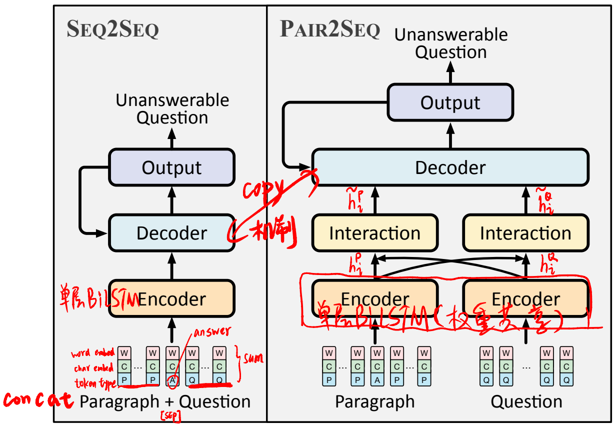

Figure 2: Diagram of the proposed pair-to-sequence model and sequence-to-sequence model. The input embeddings is the sum of the word embeddings, the character embeddings and the token type embeddings. The input questions are all answerable.

rize these work according to the type of the augmentation data: external data source, paragraphs or questions. Devlin et al. (2019) fine-tune BERT on the SQuAD dataset jointly with another dataset TriviaQA (Joshi et al., 2017). Yu et al. (2018) paraphrase paragraphs with backtranslation. Another line of work adheres to generate answerable questions. Yang et al. (2017) propose to generate questions based on the unlabeled text for semi-supervised question answering. Sun et al. (2019) propose a rule-based system to generate multiple-choice questions with candidate options upon the paragraphs. We aim at generating unanswerable questions as a means of data augmentation.

3 Problem Formulation

总目标

Given an answerable question q and its corresponding paragraph p that contains the answer a, we aim to generate unanswerable questions $\tilde{q}$ that fulfills certain requirements. First, it cannot be answered by paragraph p. Second, it must be relevant to both answerable question q and paragraph p, which refrains from producing irrelevant questions. Third, it should ask for something of the same type as answer a.

对生成的不可回答的问题的要求

As shown in Figure 2, we investigate two simple neural models built upon encoder-decoder architecture (Cho et al., 2014; Bahdanau et al., 2015) to generate unanswerable questions. A sequence-to-sequence model takes the concatenated paragraph and question as input, and encodes the input in a sequential manner. A pair-to-sequence model is further introduced to capture the interactions between inputs. The decoder of two models generates unanswerable questions sequentially. We factorize the probability of generating the unanswerable question $P(\tilde{q}|q,p,a)$ as:

$$P(\tilde{q}|q,p,a) = \prod_{t=1}^{|\tilde{q}|} P(\tilde{q}_t|\tilde{q}_{<t},q,p,a) \tag{1}$$

where $\tilde{q}_{<t} = \tilde{q}_1 \dots \tilde{q}_{t-1}$.

3.1 Sequence-to-Sequence Model

In the sequence-to-sequence model, paragraph and question pairs are packed into an ordered sequence x with a special separator in between. To indicate answers in paragraphs, we introduce token type embeddings which can also be used to distinguish questions from paragraphs in sequence-to-sequence model. As we can see in Figure 2, the token type can be answer (A), paragraph (P), or question (Q). For a given token, we construct the input representation $\mathbf{e}_i$ by summing the corresponding word embeddings, character embeddings and token type embeddings. Here characters are embedded by an embedding matrix followed by a max pooling layer.

char embedding

We apply a single-layer bi-directional recurrent neural networks with long short-term memory units (LSTM; Hochreiter and Schmidhuber, 1997) to produce encoder hidden states $\mathbf{h}_i = \mathrm{f}_{\mathrm{BiLSTM}}(\mathbf{h}_{i-1}, \mathbf{e}_i)$. On each decoding step t, the hidden states of decoder (a single-layer unidirectional LSTM network) are computed by $\mathbf{s}_t = \mathrm{f}_{\mathrm{LSTM}}(\mathbf{s}_{t-1}, [\mathbf{y}_{t-1}; \mathbf{c}_{t-1}])$, where $\mathbf{y}_{t-1}$ is the word embedding of previously predicted token and $\mathbf{c}_{t-1}$ is the encoder context vector of previous step. Besides, we use an attention mechanism to summarize the encoder-side information into $\mathbf{c}_t$ for current step. The attention distribution γ_t over source words is computed as in Luong et al. (2015):

$S_t = f_{LSTM}(S_{t-1}, [y_{t-1}; C_{t-1}])$

$$score(\mathbf{h}_i, \mathbf{s}_t) = \mathbf{h}_i^{\mathrm{T}} \mathbf{W}_\gamma \mathbf{s}_t \tag{2}$$

$$\gamma_{i,t} = \exp(score(\mathbf{h}_i, \mathbf{s}_t))/Z_t \tag{3}$$

$$\mathbf{c}_t = \sum_i^{|x|} \gamma_{i,t} \mathbf{h}_i \tag{4}$$

where $Z_t = \sum_k^{|x|} \exp(score(\mathbf{h}_k, \mathbf{s}_t))$, $\mathbf{W}_\gamma$ in score function is a learnable parameter.

Next, $\mathbf{s}_t$ is concatenated with $\mathbf{c}_t$ to produce the vocabulary distribution P_v:

$$P_v = \mathrm{softmax}(\mathbf{W}_v[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_v) \tag{5}$$

where $\mathbf{W}_v$ and $\mathbf{b}_v$ are learnable parameters. Copy mechanism (See et al., 2017) is incorporated to directly copy words from inputs, because words in paragraphs or source questions are of great value for unanswerable question generation. Specifically, we use $\mathbf{s}_t$ and $\mathbf{c}_t$ to produce a gating probability g_t:

$$g_t = \text{sigmoid}(\mathbf{W}_g[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_g) \tag{6}$$

where $\mathbf{W}_g$ and $\mathbf{b}_g$ are learnable parameters. The gate g_t determines whether generating a word from the vocabulary or copying a word from inputs. Finally, we obtain the probability of generating $\tilde{q}_t$ by:

$$P(\tilde{q}_t|\tilde{q}_{<t}, q, p, a) = g_t P_v(\tilde{q}_t) + (1 - g_t) \sum_{i \in \zeta_{\tilde{q}_t}} \hat{\gamma}_{i,t}$$

where $\zeta_{\tilde{q}_t}$ denotes all the occurrence of $\tilde{q}_t$ in inputs, and the copying score $\hat{\gamma}_t$ is computed in the same way as attention scores γ_t (see Equation (3)) while using different parameters.

3.2 Pair-to-Sequence Model

Paragraph and question interactions play a vitally important role in machine reading comprehension. The interactions make the paragraph and question aware of each other and help to predict the answer more precisely. Therefore we propose a pair-to-sequence model, conducting attention based interactions in encoder and subsequently decoding with two series of representations.

In pair-to-sequence model, the paragraph and question are embedded as in sequence-to-sequence model, but encoded separately by weight-shared bi-directional LSTM networks, yielding $\mathbf{h}_i^p = \text{f}_{\text{BiLSTM}}(\mathbf{h}_{i-1}^p, \mathbf{e}_{i-1}^p)$ as paragraph encodings and $\mathbf{h}_i^q = \text{f}_{\text{BiLSTM}}(\mathbf{h}_{i-1}^q, \mathbf{e}_{i-1}^q)$ as question encodings. The same attention mechanism as in sequence-to-sequence model is used in the following interaction layer to produce question-aware paragraph representations $\tilde{\mathbf{h}}_i^p$:

interaction
模块

$$\alpha_{i,j} = \exp(score(\mathbf{h}_i^p, \mathbf{h}_j^q))/Z_i \tag{7}$$

$$\hat{\mathbf{h}}_i^p = \sum_{j=1}^{|q|} \alpha_{i,j} \mathbf{h}_j^q \tag{8}$$

$$\tilde{\mathbf{h}}_i^p = \tanh(\mathbf{W}_p[\mathbf{h}_i^p; \hat{\mathbf{h}}_i^p] + \mathbf{b}_p) \tag{9}$$

where $Z_i = \sum_{k=1}^{|q|} \exp(score(\mathbf{h}_i^p, \mathbf{h}_k^q))$, $\mathbf{W}_p$ and $\mathbf{b}_p$ are learnable parameters. Similarly, the paragraph-aware question representations $\tilde{\mathbf{h}}_i^q$ are produced by:

$$\beta_{i,j} = \exp(score(\mathbf{h}_i^p, \mathbf{h}_j^q))/Z_j \tag{10}$$

$$\hat{\mathbf{h}}_i^q = \sum_{i=1}^{|p|} \beta_{i,j} \mathbf{h}_i^p \tag{11}$$

$$\tilde{\mathbf{h}}_j^q = \tanh(\mathbf{W}_q[\mathbf{h}_j^q; \hat{\mathbf{h}}_j^q] + \mathbf{b}_q) \tag{12}$$

where $Z_j = \sum_{k=1}^{|p|} \exp(score(\mathbf{h}_k^p, \mathbf{h}_j^q))$, $\mathbf{W}_q$ and $\mathbf{b}_q$ are learnable parameters.

Accordingly, the decoder now takes paragraph context $\mathbf{c}_{t-1}^p$ and question context $\mathbf{c}_{t-1}^q$ as encoder context, computed as $\mathbf{c}_t$ (see Equation (4)) in sequence-to-sequence model, to update decoder hidden states $\mathbf{s}_t = \text{f}_{\text{LSTM}}(\mathbf{s}_{t-1}, [\mathbf{y}_{t-1}; \mathbf{c}_{t-1}^p; \mathbf{c}_{t-1}^q])$ decoder and predict tokens. Copy mechanism is also adopted as described before, and copying words from both the paragraph and question is viable. copy机制

3.3 Training and Inference

The training objective is to minimize the negative likelihood of the aligned unanswerable question $\tilde{q}$ given the answerable question q and its corresponding paragraph p that contains the answer a:

$$\mathcal{L} = - \sum_{(\tilde{q}, q, p, a) \in \mathcal{D}} \log P(\tilde{q}|q, p, a; \theta) \tag{13}$$

loss

where $\mathcal{D}$ is the training corpus and θ denotes all the parameters. Sequence-to-sequence and pair-to-sequence models are trained with the same objective.

During inference, the unanswerable question for question answering pair (q, p, a) is obtained via $\text{argmax}_{q'} P(q'|q, p, a)$, where q' represents candidate outputs. Beam search is used to avoid iterating over all possible outputs.

4 Experiments

We conduct experiments on the SQuAD 2.0 dataset (Rajpurkar et al., 2018). The extractive machine reading benchmark contains about 100,000 answerable questions and over 50,000 crowdsourced unanswerable questions towards Wikipedia paragraphs. Crowdworkers are requested to craft unanswerable questions that are relevant to the given paragraph. Moreover, for each unanswerable question, a plausible answer span is annotated, which indicates the incorrect answer obtained by only relying on type-matching heuristics. Both answers and plausible answers are text spans in the paragraphs.

4.1 Unanswerable Question Generation

4.1.1 Training Data Construction

构造方式

We use (plausible) answer spans in paragraphs as pivots to align pairs of answerable questions and unanswerable questions. An aligned pair is shown in Figure 1. As to the spans that correspond to multiple answerable and unanswerable questions, we sort the pairs by Levenshtein distance (Levenshtein, 1966) and keep the pair with the minimum distance, and make sure that each question is only paired once.

We obtain $20,240$ aligned pairs from the SQuAD 2.0 dataset in total. The Levenshtein distance between the answerable and unanswerable questions in pairs is 3.5 on average. Specifically, the $17,475$ pairs extracted from the SQuAD 2.0 training set are used to train generation models. Since the SQuAD 2.0 test set is hidden, we randomly sample 46 articles from the SQuAD 2.0 training set with $1,805$ ($\sim$10%) pairs as holdout set and evaluate generation models with $2,765$ pairs extracted the SQuAD 2.0 development set.

4.1.2 Settings

We implement generation models upon OpenNMT (Klein et al., 2017). We preprocess the corpus with the spaCy toolkit for tokenization and sentence segmentation. We lowercase tokens and build the vocabulary on SQuAD 2.0 training set with word frequency threshold of 9 to remove most noisy tokens introduced in data collection and tokenization. We set word, character and token type embeddings dimension to 300. We use the `glove.840B.300d` pre-trained embeddings (Pennington et al., 2014) to initialize word embeddings, and do further updates during training. Both encoder and decoder share the same vocabulary and word embeddings. The hidden state size of LSTM network is 150. Dropout probability is set to 0.2. The data are shuffled and split into mini-batches of size 32 for training. The model is optimized with Adagrad (Duchi et al., 2011) with an initial learning rate of 0.15. During inference, the beam size is 5. We prohibit producing unknown words by setting the score of `<unk>` token to `-inf`. We filter the beam outputs that make no differences to the input question.

4.1.3 Evaluation Metrics

The generation quality is evaluated using three automatic evaluation metrics: BLEU (Papineni et al., 2002), ROUGE (Lin, 2004) and GLEU (Napoles et al., 2015). BLEU[1] is a commonly used metric in machine translation that computes n-gram precisions over references. Recall-oriented ROUGE[2] metric is widely adopted in summarization, and ROUGE-L measures longest common subsequence between system outputs and references. GLEU[3] is a variant of BLEU with the modification that penalizes system output n-grams that present in input but absent from the reference. This makes GLEU a preferable metric for tasks with subtle but critical differences in a monolingual setting as in our unanswerable question generation task.

We also conduct human evaluation on 100 samples in three criteria: (1) unanswerability, which indicates whether the question is unanswerable or not; (2) relatedness, which measures semantic relatedness between the generated question and input question answering pair; (3) readability, which indicates the grammaticality and fluency. We ask three raters to score the generated questions in terms of relatedness and readability on a 1-3 scale (3 for the best) and determine the answerability in binary (1 for unanswerable). The raters are not aware of the question generation methods in advance.

4.1.4 Results

Results of the automatic evaluation are shown in Table 1. We find that the proposed pair-to-sequence model that captures interactions between paragraph and question performs consistently better than sequence-to-sequence model. Moreover, replacing the input paragraph with the answer sentence hurts model performance, which indicates that using the whole paragraph as context provides more helpful information to unanswerable question generation. We also try to generate unanswerable questions by only relying on answerable questions (see "-Paragraph"), or the paragraph (see "-Question"). Unsurprisingly, both ablation models obtain worse performance compared with the full model. These two ablation results also demonstrate that the input answerable question helps more to improve performance compared with the input paragraph. We argue that the original answerable question provides more direct information due to the fact that the average edit distance between the example pairs is 3.5. At last,

[1] `github.com/moses-smt/mosesdecoder`
[2] `pypi.org/project/pyrouge`
[3] `github.com/cnap/gec-ranking`

Model	GLEU-3	GLEU-4	BLEU-3	BLEU-4	ROUGE-2	ROUGE-3	ROUGE-L
Seq2Seq	33.13	27.39	36.80	27.84	46.54	32.98	64.28
Pair2Seq	**35.06**	**29.43**	**37.67**	**29.17**	**47.46**	**34.18**	**65.24**
- Paragraph (+AS)	34.42	28.43	37.35	28.44	47.13	33.29	65.02
- Paragraph	33.58	27.54	35.89	26.99	46.14	31.45	64.78
- Question	9.40	6.21	6.7	3.1	12.64	5.63	32.26
- Copy	25.06	19.80	36.06	22.84	33.40	20.45	52.76

Table 1: Automatic evaluation results. Higher score is better and the best performance for each evaluation metric is highlighted in **boldface**. "- Paragraph (+AS)" represents replacing paragraphs with answer sentences.

	EM / F1	△
BNA	59.7/62.7	-
+ UnAnsQ	61.0/63.5	+1.3/+0.8
DocQA	61.9/64.5	-
+ UnAnsQ	62.4/65.3	+0.5/+0.8
BERT$_{Base}$	74.3/77.4	-
+ UnAnsQ	76.4/79.3	+2.1/+1.9
BERT$_{Large}$	78.2/81.3	-
+ UnAnsQ	80.0/83.0	+1.8/+1.7

Table 2: Experimental results of applying data augmentation to reading comprehension models on the SQuAD 2.0 dataset. "△" indicates absolute improvement.

	UnAns	Rela	Read
TfIdf	**0.96**	1.52	**2.98**
Seq2Seq	0.62	2.88	2.39
Pair2Seq	0.65	**2.95**	2.61
Human	0.95	2.96	3

Table 3: Human evaluation results. Unanswerability (UnAns): 1 for unanswerable, 0 otherwise. Relatedness (Rela): 3 for relevant to both answerable question and paragraph, 2 for relevant to only one, 1 for irrelevant. Readability (Read): 3 for fluent, 2 for minor grammatical errors, 1 for incomprehensible.

Type	S2S	P2S	Human
Negation	42%	54%	32%
Antonym	4%	5%	9%
Entity Swap	17%	20%	20%
Mutual Exclusion	2%	0%	12%
Impossible Condition	8%	12%	25%
Other	27%	8%	2%

Table 4: Types of unanswerable questions generated by models and humans, we refer the reader to (Rajpurkar et al., 2018) for detail definition of each type. "S2S" represents the sequence-to-sequence baseline and "P2S" is our proposed pair-to-sequence model.

we remove the copy mechanism that restrains prediction tokens to the vocabulary. The results indicate the necessity of copying tokens from answerable questions and paragraphs to outputs, which relieves the out-of-vocabulary problem.

Table 3 shows the human evaluation results of generated unanswerable questions. We compare with the baseline method TfIdf, which uses the input answerable question to retrieve similar questions towards other articles as outputs. The retrieved questions are mostly unanswerable and readable, but they are not quite relevant to the question answering pair. Notice that being relevant is demonstrated to be important for data augmentation in further experiments on machine reading comprehension. Here pair-to-sequence model still outperforms sequence-to-sequence model in terms of all three metrics. But the differences in human evaluation are not as notable as in the automatic metrics.

As shown in Table 4, we further randomly sample 100 system outputs to analyze the types of generated unanswerable questions. We borrow the types defined in Rajpurkar et al. (2018) for SQuAD 2.0. We categorize the outputs with grammatical errors that make them hard to understand into `Other`. Samples that fall into `Impossible Condition` are mainly produced by non-entity substitution. We can see that models tend to generate unanswerable questions by inserting negation and swapping entities. These two types are also most commonly used when crowdworkers pose unanswerable questions according to answerable ones. We also find that the current models still have difficulties in utilizing antonyms and exclusion conditions, which could

Title: Victoria (Australia)
Paragraph: Victorian schools are either publicly or privately funded. Public schools, also known as state or government schools, are funded and run directly by the Victoria Department of Education . Students do not pay tuition fees, but some extra costs are levied. Private fee-paying schools include parish schools run by the Roman Catholic Church and independent schools similar to British public schools. Independent schools are usually affiliated with Protestant churches. Victoria also has several private Jewish and Islamic primary and secondary schools. Private schools also receive some ...

Question: What organization runs the public schools in Victoria?
Human: What organization runs the waste management in Victoria?
Seq2Seq: what organization runs the public schools in texas ?
Pair2Seq: what organization never runs the public schools in victoria ?
(Plausible) Answer: Victoria Department of Education

Question: What church runs some private schools in Victoria?
Human: What church runs public schoolsin Victoria?
Seq2Seq: what church runs some private schools ?
Pair2Seq: what church no longer runs some private schools in victoria ?
(Plausible) Answer: Roman Catholic Church

Question: Since students do not pay tuition, what do they have to pay for schooling in Victoria?
Human: What is covered by the state in addition to tuition?
Seq2Seq: since students do not pay to pay schooling in victoria ?
Pair2Seq: since students do n't pay tuition , what do they have to pay for schooling in victoria ?
(Plausible) Answer: some extra costs

Question: What are public schools in Victoria?
Human: What are public banks in Victoria?
Seq2Seq: what are n't public schools in victoria ?
Pair2Seq: what are public schools not in victoria ?
(Plausible) Answer: state or government schools

Figure 3: Sample output generated by human, sequence-to-sequence model, and pair-to-sequence model. The (plausible) answer span of questions are marked in colors and main difference of model outputs are underlined.

be improved by incorporating external resources.

In Figure 3, we present a sample paragraph and its corresponding answerable questions and generated unanswerable questions. In the first example, two models generate unanswerable questions by swapping the location entity "*Victoria*" with "*texas*" and inserting negation word "*never*", respectively. In the second example, sequence-to-sequence model omits the condition "*in Victoria*" and yields an answerable question. Pair-to-sequence model inserts the negation "*no longer*" properly, which is not mentioned in the paragraph. In the third example, grammatical errors are found in the output of Seq2Seq. The last example shows that inserting negation words in different positions ("*n't public*" versus "*not in victoria*") can express different meanings. Such cases are critical for generated questions' answerability, which is hard to handle in a rule-based system.

4.2 Data Augmentation for Machine Reading Comprehension

4.2.1 Question Answering Models

We apply our automatically generated unanswerable questions as augmentation data to the following reading comprehension models:

BiDAF-No-Answer (BNA) BiDAF (Seo et al., 2017) is a benchmark model on extractive machine reading comprehension. Based on BiDAF, Levy et al. (2017) propose the BiDAF-No-Answer model to predict the distribution of answer candidates and the probability of a question being unanswerable at the same time.

DocQA Clark and Gardner (2018) propose the DocQA model to address document-level reading comprehension. The no-answer probability is also predicted jointly.

BERT Fine-Tuning It is the state-of-the-art model on unanswerable machine reading comprehension. We adopt the uncased version of BERT (Devlin et al., 2019) for fine-tuning. The batch sizes of BERT-base and BERT-large are set to 12 and 24 respectively. The rest hyperparameters are kept untouched as in the official instructions of fine-tuning BERT-Large on SQuAD 2.0.

4.2.2 Data Augmentation Setup

We first generate unanswerable questions using the trained generation model. Specifically, we use

	EM / F1	△
BERT$_{Base}$	74.3/77.4	-
+ TFIDF	75.0/77.8	+0.7/+0.4
+ RULE	75.6/78.5	+1.3/+1.1
+ SEQ2SEQ	75.5/78.2	+1.2/+0.8
+ PAIR2SEQ	76.4/79.3	+2.1/+1.9

Table 5: Results using different generation methods for data augmentation. "△" indicates absolute improvement.

the answerable questions in the SQuAD 2.0 training set, besides ones aligned before, to generate unanswerable questions. Then we use the paragraph and answers of answerable questions along with the generated questions to construct training examples. At last, we have an augmentation data containing $69,090$ unanswerable examples.

We train question answering models with augmentation data in two separate phases. In the first phase, we train the models by combining the augmentation data and all $86,821$ SQuAD 2.0 answerable examples. Subsequently, we use the original SQuAD 2.0 training data alone to further fine-tune model parameters.

4.2.3 Results

Exact Match (EM) and F1 are two metrics used to evaluate model performance. EM measures the percentage of predictions that match ground truth answers exactly. F1 measures the word overlap between the prediction and ground truth answers. We use pair-to-sequence model with answerable questions and paragraphs for data augmentation by default.

Table 2 shows the exact match and F1 scores of multiple reading comprehension models with and without data augmentation. We can see that the generated unanswerable questions can improve both specifically designed reading comprehension models and strong BERT fine-tuning models, yielding 1.9 absolute F1 improvement with BERT-base model and 1.7 absolute F1 improvement with BERT-large model. Our submitted model obtains an EM score of 80.75 and an F1 score of 83.85 on the hidden test set.

As shown in Table 5, pair-to-sequence model proves to be a better option for generating augmentation data than other three methods. Besides the sequence-to-sequence model, we use answerable questions to retrieve questions from other articles with TFIDF. The retrieved questions are of little help to improve the model, because they are less relevant to the paragraph as shown in Table 3. We refer to the rule-based method (Jia and Liang, 2017) that swaps entities and replaces words with antonyms as RULE. In comparison to the above methods, pair-to-sequence model can yield the largest improvement.

	EM / F1	△
BERT$_{Base}$	74.3/77.4	-
+ UNANSQ×1	76.4/79.3	+2.1/+1.9
+ UNANSQ×2	76.4/79.4	+2.1/+2.0
+ UNANSQ×3	76.6/79.6	+2.3/+2.2
BERT$_{Large}$	78.2/81.3	-
+ UNANSQ×1	80.0/83.0	+1.8/+1.7
+ UNANSQ×2	80.0/82.9	+1.8/+1.6
+ UNANSQ×3	80.1/83.1	+1.9/+1.8

Table 6: Ablation over the size of data augmentation. "× N" means the original size is enhanced N times. "△" indicates absolute improvement.

Results in Table 6 show that enlarging the size of augmentation data can further improve model performance, especially with the BERT-base model. We conduct experiments using two and three times the size of the base augmentation data (i.e., $69,090$ unanswerable questions). We generate multiple unanswerable questions for each answerable question by using beam search. Because we only generate unanswerable questions, the data imbalance problem could mitigate the improvement of incorporating more augmentation data.

5 Conclusions

In this paper, we propose to generate unanswerable questions as a means of data augmentation for machine reading comprehension. We produce relevant unanswerable questions by editing answerable questions and conditioning on the corresponding paragraph. A pair-to-sequence model is introduced in order to capture the interactions between question and paragraph. We also present a way to construct training data for unanswerable question generation models. Both automatic and human evaluations show that the proposed model consistently outperforms the sequence-to-sequence baseline. The results on the SQuAD 2.0 dataset show that our generated unanswerable questions can help to improve multiple reading comprehension models. As for future work, we would like to enhance the ability to utilize antonyms for unanswerable question generation by leveraging external resources.

Acknowledgments

We thank anonymous reviewers for their helpful comments. Qin and Liu were supported by National Natural Science Foundation of China (NSFC) via grants 61632011 and 61772156.

References

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proceedings of the 3rd International Conference on Learning Representations*.

Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734. Association for Computational Linguistics.

Christopher Clark and Matt Gardner. 2018. Simple and effective multi-paragraph reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 845–855. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics.

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. *Computing Research Repository*, arXiv:1905.03197. Version 1.

Xinya Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from wikipedia. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1907–1917. Association for Computational Linguistics.

Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1342–1352. Association for Computational Linguistics.

John Duchi, Elad Hazan, and Yoram Singer. 2011. Adaptive subgradient methods for online learning and stochastic optimization. *Journal of Machine Learning Research*, 12(Jul):2121–2159.

Jiatao Gu, Zhengdong Lu, Hang Li, and Victor O.K. Li. 2016. Incorporating copying mechanism in sequence-to-sequence learning. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1631–1640. Association for Computational Linguistics.

Michael Heilman and Noah A. Smith. 2010. Good Question! Statistical ranking for question generation. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 609–617. Association for Computational Linguistics.

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

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

Minghao Hu, Yuxing Peng, Zhen Huang, Xipeng Qiu, Furu Wei, and Ming Zhou. 2018. Reinforced mnemonic reader for machine reading comprehension. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4099–4106. International Joint Conferences on Artificial Intelligence Organization.

Hsin-Yuan Huang, Chenguang Zhu, Yelong Shen, and Weizhu Chen. 2018. FusionNet: Fusing via fully-aware attention with application to machine comprehension. In *International Conference on Learning Representations*.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2021–2031. Association for Computational Linguistics.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual*

Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 1601–1611. Association for Computational Linguistics.

Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *AAAI Conference on Artificial Intelligence*.

Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander Rush. 2017. OpenNMT: Open-source toolkit for neural machine translation. In *Proceedings of ACL 2017, System Demonstrations*, pages 67–72. Association for Computational Linguistics.

Tomas Kocisky, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gabor Melis, and Edward Grefenstette. 2018. The NarrativeQA reading comprehension challenge. *Transactions of the Association for Computational Linguistics*, 6:317–328.

Vladimir I Levenshtein. 1966. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet Physics Doklady*, 10(8):707–710.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 333–342. Association for Computational Linguistics.

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

Xiaodong Liu, Yelong Shen, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for machine reading comprehension. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1694–1704. Association for Computational Linguistics.

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421. Association for Computational Linguistics.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 588–593. Association for Computational Linguistics.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. MS MARCO: A human generated machine reading comprehension dataset. *Computing Research Repository*, arXiv:1611.09268. Version 3.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Jeffrey Pennington, Richard Socher, and Christopher 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. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392. Association for Computational Linguistics.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083. Association for Computational Linguistics.

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

Linfeng Song, Zhiguo Wang, Wael Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 569–574. Association for Computational Linguistics.

Kai Sun, Dian Yu, Dong Yu, and Claire Cardie. 2019. Improving machine reading comprehension with general reading strategies. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3930–3939. Association for Computational Linguistics.

Chuanqi Tan, Furu Wei, Qingyu Zhou, Nan Yang, Weifeng Lv, and Ming Zhou. 2018. I know there is no answer: Modeling answer validation for machine reading comprehension. In *Natural Language Processing and Chinese Computing*, pages 85–97, Cham. Springer International Publishing.

Shuohang Wang and Jing Jiang. 2017. Machine comprehension using Match-LSTM and answer pointer. In *International Conference on Learning Representations*.

Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1705–1714. Association for Computational Linguistics.

Wenhui Wang, Nan Yang, Furu Wei, Baobao Chang, and Ming Zhou. 2017. Gated self-matching networks for reading comprehension and question answering. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 189–198. Association for Computational Linguistics.

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

Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017. Semi-supervised QA with generative domain-adaptive nets. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1040–1050. Association for Computational Linguistics.

Kaichun Yao, Libo Zhang, Tiejian Luo, Lili Tao, and Yanjun Wu. 2018. Teaching machines to ask questions. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*, pages 4546–4552. International Joint Conferences on Artificial Intelligence Organization.

Wen-tau Yih, Ming-Wei Chang, Christopher Meek, and Andrzej Pastusiak. 2013. Question answering using enhanced lexical semantic models. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1744–1753. Association for Computational Linguistics.

Adams Wei Yu, David Dohan, Quoc Le, Thang Luong, Rui Zhao, and Kai Chen. 2018. Fast and accurate reading comprehension by combining self-attention and convolution. In *International Conference on Learning Representations*.

Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine comprehension by text-to-text neural question generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 15–25. Association for Computational Linguistics.

Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2018. Neural question generation from text: A preliminary study. In *Natural Language Processing and Chinese Computing*, pages 662–671, Cham. Springer International Publishing.