Answer-Supervised Question Reformulation for Enhancing Conversational Machine Comprehension

Qian Li[◇*], **Hui Su**[‡], **Cheng Niu**[‡], **Daling Wang**[◇], **Zekang Li**[♠], **Shi Feng**[◇], **Yifei Zhang**[◇]
[◇]School of Computer Science and Engineering, Northeastern University, Shenyang, China
[‡]Pattern Recognition Center, WeChat AI, Tencent Inc, China
[♠]Key Laboratory of Intelligent Information Processing
Institute of Computing Technology, Chinese Academy of Sciences
qianli@stumail.neu.edu.cn, {wangdaling,fengshi,zhangyifei}@cse.neu.edu.cn,
{aaronsu,chengniu}@tencent.com, lizekang19g@ict.ac.cn

Abstract

In conversational machine comprehension, it has become one of the research hotspots integrating conversational history information through question reformulation for obtaining better answers. However, the existing question reformulation models are trained only using supervised question labels annotated by annotators without considering any feedback information from answers. In this paper, we propose a novel **A**nswer-**S**upervised **Q**uestion **R**eformulation (ASQR) model for enhancing conversational machine comprehension with reinforcement learning technology. ASQR utilizes a pointer-copy-based question reformulation model as an **agent**, takes an **action** to predict the next word, and observes a **reward** for the whole sentence **state** after generating the end-of-sequence token. The experimental results on QuAC dataset prove that our ASQR model is more effective in conversational machine comprehension. Moreover, pretraining is essential in reinforcement learning models, so we provide a high-quality annotated dataset for question reformulation by sampling a part of QuAC dataset.

1 Introduction

The performance of the single-turn machine comprehension models has been greatly improved, even close to human-level recently (Wang et al., 2018; Devlin et al., 2018; Sun et al., 2018; Hu et al., 2018; Liu et al., 2017), while the conversational machine comprehension models are far from satisfactory (Choi et al., 2018; Huang et al., 2018; Zhu et al., 2018). In single-turn machine comprehension, different questions for the same paragraph have no connection. However, the questions omitting a great of key information in conversational machine comprehension are only meaningful by considering the previous questions and answers history (Table 1). Therefore, the major difficulty of solving conversational machine comprehension lies in how to integrate the conversational history when answering the questions.

Sentence reformulation aims to get more fluent and meaningful sentences based on supplementary information (Liu et al., 2018; Rastogi et al., 2019), and has been adopted in abstract extraction (Nallapati et al., 2016; See et al., 2017), query reformulation (Riezler and Liu, 2010; Rastogi et al., 2019), and translation reformulation (Niehues et al., 2016; Junczys-Dowmunt and Grundkiewicz, 2017). Question reformulation (Buck et al., 2017; Nogueira and Cho, 2017; Rastogi et al., 2019), as an important branch of sentence reformulation, aims to reformulate question according to conversational history.

However, the existing question reformulation models are trained with annotated labels via a training mechanism as *teacher forcing* (Bengio et al., 2015). The annotated labels-supervised training approaches have some drawbacks: **(1) Minority**: Due to the limitation of human resources and funds, annotated data only accounts for a small part of all data. **(2) Errors**: Some fatal errors that adversely affect model training may exist in annotated data inadvertently. **(3) Unmet requirements**: What deserves attention is that the training mechanism for the existing question reformulation models do not consider any feedback information from subsequent functions, while the feedback information is always important. Particularly, the question reformulation model in conversational machine comprehension aims to get better answers, so the quality of the reformulated questions should depend on gold answers but not question labels. To our best knowledge, there are some preliminary attempts to reformulate question with downstream feedback in question answering

[*]This work was done when Qian Li was interning at Pattern Recognition Center, WeChat AI, Tencent.

Title: Skid Row	
Paragraph: Skid Row, released in January 1989, was an instant success. The record went 5x platinum on the strength of the Top 10 singles. Skid Row supported the album by opening for Bon Jovi on their New Jersey tour. As part of the six-month tour, Skid Row played its first ever UK gig supporting Bon Jovi's outdoor show at Milton Keynes Bowl on August 19, 1989. ... CANNOTANSWER.	
Q1: Did they release any albums	A1: Skid Row, released in January 1989
Q2: How did it do **Q2': How did Skid Row do**	A2: instant success
Q3: Did it go on tour **Q3': Did Skid Row go on tour**	A3: first supporting Bon Jovi's outdoor show
Q4: Did the Tour have a name **Q4': Did the outdoor show have a name**	A4: New Jersey tour
Q5: How long did the tour last **Q5': How long did the New Jersey tour last**	A5: CANNOTANSWER

Table 1: An example of conversational machine comprehension from QuAC dataset (Choi et al., 2018). Giving a paragraph title, the student asks teacher questions according to the conversational history. The teacher answers the question by choosing a text span from the paragraph context or CANNOTANSWER. **Qi'** is the reformulated question for Qi by annotators.

tasks (Buck et al., 2017; Nogueira and Cho, 2017), while no work in conversational machine comprehension tasks. How to train the question reformulation models with supervised information from answers in conversational machine comprehension is still a major challenge.

In this paper, we present ASQR, an Answer-Supervised Question Reformulation model for conversational machine comprehension with reinforcement learning technology (Figure 1). At our ASQR model, the agent, a novel pointer-copy-based question reformulation model proposed in Section 2, takes an action to predict the next word. The state for the whole sentence is composed of continuous actions and end with the end-of-sequence (EOS) signal. The agent only observes a reward for the whole sentence state after generating the EOS token, which is quite different from the teacher forcing models. The reward is the similarity score between the gold answer and the predicted answer obtained by feeding the whole sentence state to a single-turn machine comprehension model.

We validate the effectiveness of our ASQR model on QuAC dataset (Choi et al., 2018). Pre-training is essential in deep reinforcement learning models (Yin et al., 2018; Xiong et al., 2018), so we sample a part of QuAC dataset, and reformulate the questions according to the conversational history by several professional annotators. The major contributions of this paper are as follows:

- We present a novel answer-supervised question reformulation model for conversational machine comprehension with reinforcement learning technology, which could be a new study direction for conversational problems.
- We provide a high-quality annotated dataset for question reformulation in conversational comprehension, which could be of great help to future related research.
- The experimental results outperforming the baseline models on the benchmark dataset prove that our model is more effective in conversational machine comprehension.

In Section 2, we will present a new pointer-copy-based question reformulation model which is as an agent in the ASQR model. The overall ASQR model with reinforcement learning technology is presented in Section 3. Then in Section 4, we introduce our annotated dataset and the experiments. The related work and some conclusions are drawn in Section 5 and 6.

2 Question Reformulation Model

In this section, we present a novel question reformulation model based on the pointer copy mechanism, which is the agent of our ASQR model in Section 3. The question reformulation model is an encoder-decoder framework shown in the left of Figure 1. The encoder is to encode the questions and their conversational history separately with the recurrent neural network. The decoder,

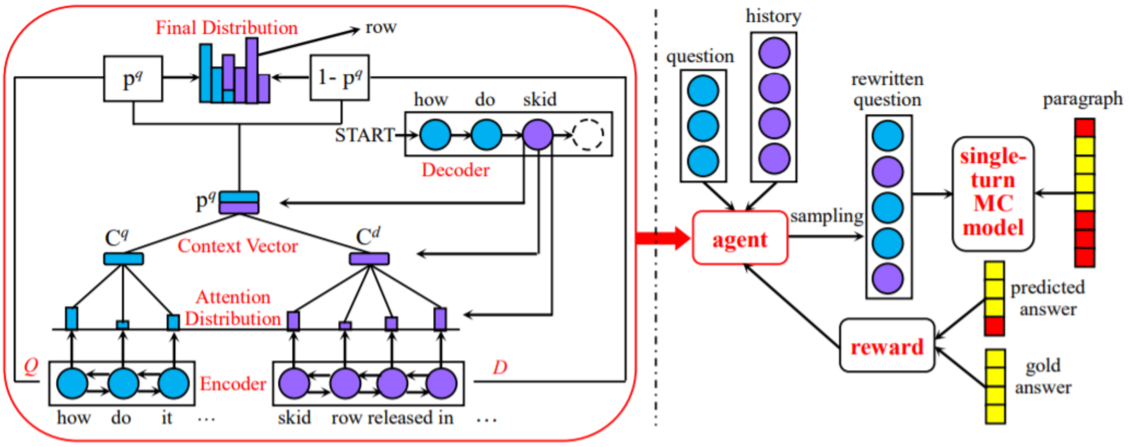

Figure 1: Our proposed ASQR Model. The left is our pointer-copy-based question reformulation model. The right is the overall perspective of the ASQR model with the left model as an agent.

a copy mechanism, copies a word from questions or conversational history according to a gate network at each time step. For simplicity, we denote each training sample as (D, Q, R), therein $D = \{Q_1, A_1, ..., Q_{n-1}, A_{n-1}\}$ represents the conversational history, (Q_i, A_i) represents the question and answer in the ith turn of the conversation, $Q = Q_n$ is the question in nth turn of the conversation. R is the reformulated question carrying important conversational information for the question Q.

2.1 Encoder

The role of the Encoder is to get the representation for the input sentence. There are two types of the input sentence: question $Q = \{x_1^q, ..., x_{m_q}^q\}$ and its conversational history $D = \{x_1^d, ..., x_{m_d}^d\}$, m_q, m_d are the number of words in question and conversational history. Here we employ bidirectional LSTM (BiLSTM) to encode each word in the sentence(Lee et al., 2017), where the BiLSTM is defined as:

$$h_t^q = BiLSTM(x_t^q) \tag{1}$$

$$h_t^d = BiLSTM(x_t^d) \tag{2}$$

where h_t^q is the representation for the word x_t^q in the question sentence, h_t^d is the representation for the word x_t^d in the conversational history sentence.

2.2 Decoder

The Decoder is to generate the reformulated questions based on the representation of questions and conversational history sentence in the Encoder. The essence of the Decoder is a copy mechanism. Decoder copies words from the input question Q or the input conversational history D. For each training sample, we should retain the original key information from the input question, and replace pronouns with entities in the conversational history, and get complete information from the conversational history if the question is incomplete.

At each time step t, let s_t be the decoder hidden state, the context vector of question be c_t^q, the context vector of conversational history be c_t^d, and the output word be y_t. The hidden state s_t can be constructed by the LSTM function as follows:

$$s_t = LSTM(s_{t-1}, c_{t-1}^q, c_{t-1}^d, y_{t-1}) \tag{3}$$

$$s_0 = tanh(W_0^q h_1^q + W_0^d h_1^d + b) \tag{4}$$

where the initial state s_0 is obtained by an activation function, W_0^q, W_0^d, b are learnable parameters.

The context vector c_t^q, c_t^d for the time step t can be computed by the attention mechanism(Luong et al., 2015; Zhou et al., 2018). We use the decoder hidden state s_t and the representation of input sentence from the encoder to get an importance score. Especially, the context vector c_t^q of question is:

$$e_{t,i} = v^T tanh(W s_t + U h_i^q) \tag{5}$$

$$a_{t,i} = \frac{exp(e_{t,i})}{\sum_{i=1}^{m_q} exp(e_{t,i})} \tag{6}$$

$$c_t^q = \sum_{i=1}^{m_q} a_{t,i} h_i^q \tag{7}$$

where v, W, U are all learnable parameters. For simplicity, we define the above attention as $c_t^q = Atten(s_t, h_i^q)$. When computing the context vector c_t^d of conversational history, it is necessary to consider the context vector of question. Therefore, the context vector c_t^d of conversational history is:

$$c_t^d = Atten((s_t, c_t^q), h_i^d) \tag{8}$$

Next, we present a switch gate network to decide to copy words from questions or conversational history. The switch gate network can be obtained based on the embedding of the previous output word y_{t-1}, the current hidden state s_t and the current context vector c_t^q, c_t^d (Zhou et al., 2018).

$$p_t^q = \sigma(w_t^y y_{t-1} + w_t^s s_t + w_t^q c_t^q + w_t^d c_t^d + b) \tag{9}$$

$$p_t^d = 1 - p_t^q \tag{10}$$

where σ is a sigmoid activation function, p_t^q is the probability of copying a word from the questions, and p_t^d is the probability of copying a word from the conversational history at the time step t.

After determining the source (input question or conversational history) of the copying words, we need to design the location of each copying word. Here, we use the pointer network (PtrN) (Vinyals et al., 2015; Zhou et al., 2018) to get the attention distribution of the words in the input questions and conversation history separately.

$$k_{i,t}^q = PtrN(s_t, h_i^q) \tag{11}$$

$$k_{i,t}^d = PtrN(s_t, h_i^d) \tag{12}$$

Therefore, we can get the probability of a word ν copying from the input question P_q and from the conversational history P_d:

$$P_q(y_t = \nu) = p_t^q * k_{\nu,t}^q \tag{13}$$

$$P_d(y_t = \nu) = p_t^d * k_{\nu,t}^d = (1 - p_t^q) * k_{\nu,t}^d \tag{14}$$

$$\begin{aligned} P(y_t = \nu) &= P_q(y_t = \nu) + P_d(y_t = \nu) \\ &= p_t^q * k_{\nu,t}^q + (1 - p_t^q) * k_{\nu,t}^d \end{aligned} \tag{15}$$

2.3 Pretrained Question Reformulation

Pretraining is essential in deep reinforcement learning(Yin et al., 2018; Xiong et al., 2018), so we pretrain the question reformulation model with the annotated data. The objective of the question reformulation model is to minimize the negative log-likelihood loss $L(\theta)$:

$$L(\theta) = -\frac{1}{N}\sum_{i=1}^{N}\sum_{t=1}^{T}\log P(y_t) \tag{16}$$

where N is the number of the training dataset, y be the annotated question for the input question Q, and T is the number of the words in y.

3 Overall ASQR Model

In this section, we introduce our proposed answer-supervised question reformulation model ASQR for conversational machine comprehension as shown in Figure 1. The architecture of our ASQR model is a reinforcement learning framework with the question reformulation model in Section 2 as an agent. In a conversational machine comprehension example, ASQR first reformulates the input questions by question reformulation model, then feeds the reformulated questions to a single-turn machine comprehension model and gets the predicted answers. The similarity scores between predicted answers and gold answers are as the reward to optimize the question reformulation model. The details are as follows:

Agent: The question reformulation model in Section 2 is defined as the agent. The reinforcement learning agent is a policy network $\pi_\theta(state, action) = p_\theta(action|state)$, where θ represents the model's parameters.

Action: The action is to predict the next word y_t by the agent. The word y_t is sampled from the input question, or from the input conversational history according to the probability distribution of vocabulary.

State: After each action, the state is updated by the agent. The state of the whole sentence is defined as $S_T = (y_1, ..., y_T)$, where y_t is the action in the time step t, T is the number of words in the sentence, and the last action y_T is an end-of-sequence token.

Reward: For each state S_T, the agent observes a reward. At this, we feed the state S_T to a pretrained single-turn machine comprehension model. The pretrained single-turn machine comprehension model predicts the answer for the state S_T, and computes the similarity score between the predicted answer and the gold answer. The similarity score is as the reward $R(S_T)$.

The goal of our reinforcement learning is to train the parameters of the agent. At this, we use the REINFORCE policy gradient algorithm (Williams, 1992; Keneshloo et al., 2018) to minimize the negative expected reward.

$$J(\theta) = -E_{S_T \sim p_\theta} R(S_T) \tag{17}$$

Because the expectation is exponential in the length of the action sequence, it always gets an unbiased estimate of the gradient instead of the full gradient. The expected gradient can be estimated with a single sample $S_T \sim p_\theta$. So the expected gradient of a non-differentiable reward function is as follows:

$$\begin{aligned}\nabla_\theta J(\theta) &= -\nabla_\theta E_{S_T \sim p_\theta} R(S_T) \\ &= -E_{S_T \sim p_\theta} \nabla_\theta \log p_\theta(S_T) R(S_T) \\ &\approx -\nabla_\theta \log p_\theta(S_T) R(S_T)\end{aligned} \tag{18}$$

But the variance for estimation of the gradient may be very high, which makes the results difficult to observe. Steven et al. (Rennie et al., 2016) prove that subtracting a baseline value from the reward $R(S_T)$ does not change the expected gradient if the baseline value does not depend on the action. Therefore, we can subtract a baseline value to reduce the variance, and the baseline can be an arbitrary action-independent function. If the reward for an action is greater than baseline, the action will be encouraged, otherwise discouraged. Here, the baseline $R(S_T^g)$ we used is the output sentence of our question reformulation model by a greedy search(Rennie et al., 2016). The expected gradient of the reward function is:

$$\nabla_\theta J(\theta) \approx -\nabla_\theta \log p_\theta(S_T)(R(S_T) - R(S_T^g)) \tag{19}$$

Using the chain rule, the above equation can be reformulated as:

$$\nabla_\theta J(\theta) = \sum_{t=1}^{T} \frac{\partial J(\theta)}{\partial o_t} \frac{\partial o_t}{\theta} \tag{20}$$

where o_t is the input to the softmax function. The gradient of $\frac{\partial J(\theta)}{\partial o_t}$ is given by (Rennie et al., 2016; Keneshloo et al., 2018):

$$\frac{\partial J(\theta)}{\partial o_t} \approx (p_\theta(y_t|h_t) - 1(y_t))(R(S_T) - R(S_T^g)) \tag{21}$$

Pretrained Single-turn MC Model In our model, the agent observes a reward for each sentence state S_T, so we need a pretrained single-turn machine comprehension model to return a reward. The single turn machine comprehension model we used is the Bert model with one additional output layer(Devlin et al., 2018), which has been proved to do well on the single-turn SQuAD dataset (Rajpurkar et al., 2018).

Type	dataPretrain			QuAC		
	train	val	test	train	val	test
questions	20k	5k	3k	81k	7k	7k
dialogs	3k	600	400	11k	1k	1k

Table 2: data statistics.

4 Experiments

In the following work, we evaluate our model on QuAC dataset(Choi et al., 2018). To prove the performance of the model, we will conduct experiments from two perspectives: (1) Quality of the question reformulation model: How our question reformulation model in Section 2 can reformulate question accurately. (2) Effectiveness of the ASQR model: whether the reformulated questions by our ASQR model are more effective in conversational machine comprehension.

4.1 Dataset

We use the QuAC dataset (Choi et al., 2018) to evaluate our model. Table 1 gives an example of conversational machine comprehension in QuAC dataset. In this conversational machine comprehension data, students ask teachers questions based on the conversational history, teachers answer the questions by intercepting fragments from the context or cannot answer. For experiments, there are two types of dataset: (1) **dataPretrain**: Our annotated dataset to pretrain the question reformulation model in section 2. (2) **QuAC**: The all official QuAC dataset to train our ASQR model.

Our annotated data **dataPretrain** with 28k questions and 4k dialogs have been sampled from QuAC dataset randomly and annotated through a formal annotation platform. Annotators reformulate question earnestly according to the conversational history if at least one of coreference and omission occurs in current question. In the case of sentence fluency, annotators only copy words, but can not introduce extra words. To ensure the annotation quality, 15% of annotated questions are daily examined by a manager, and considered acceptable when the accuracy surpasses 90%. Some annotated questions can be seen in Table 1.

The investigation on our annotated dataset shows that there are 51.7%-coreference and 10.1%-omission questions, only 38.2% questions don't need to reformulated, which proves that

Model	BLEU1	BLEU2	BLEU3	BLEU4	EM	ROUGE_L	F1
Generate	56.18	47.38	37.01	27.43	11.09	62.65	66.36
Ptr-Generate	76.02	71.83	66.53	61.64	45.93	81.97	83.73
Ptr-Net	76.75	72.72	67.83	62.15	47.20	82.47	84.12
Ptr-Copy(4-qa)	78.13	73.84	68.20	62.52	47.20	83.49	85.22
Ptr-Copy(all-qa)	**78.74**	**74.80**	**69.67**	**64.20**	**49.85**	**84.15**	**85.75**

Table 3: BLEU-1,2,3,4, EM, ROUGE_L and F1 scores on the test dataset in the dataPretrain.

question reformulation is necessary and important for downstream tasks. We divide the dataPretrain dataset into a training dataset (7/10), a validation dataset (2/10), a test dataset (1/10). Table 2 describes the data statistics.

4.2 Settings

Question Reformulation Model We train the question reformulation model with the loss in Section 2.3 and the annotated **dataPretrain**. We built our vocabulary based on the nltk word tokenizer for all QuAC dataset. The vocabulary size we used is 10697. We set the word embedding as 128. The dimension of hidden states for both encoder and decoder is 256. The batch size is 64. The max encoder step is 400, the max decoder step is 30, and the minimum decoder steps is 5. We use Adagrad to train our model, wherein the learning rate is 0.1 and the initial accumulator value is 0.1. In the test stage, we generate reformulated question by the beam search strategy, the beam size is 4.

Pretrained Single turn MC Model We use the Bert model with one additional output layer (Devlin et al., 2018) as our single-turn machine comprehension model, which has a good performance on SQuAD2.0 dataset. The pretrained model of Bert we used is *BERT-Base, Uncased* with 12 layers, 768 hidden states, 12 heads and 110M parameters. The batch size is 24. The maximum length of an answer that can be generated is 30. The initial single-turn machine comprehension model is fine-tuned with all official QuAC data. If the reformulated questions are more meaningful than official questions, we will fine-tune the single-turn machine comprehension model with the reformulated data. The parameters of the single-turn machine comprehension model are fixed when training our ASQR model.

ASQR Model Our ASQR model can be trained based on above pretrained question reformulation model and single-turn machine comprehension model. We use the Adam optimizer with 1e-5 learning rate to update the trainable parameters in our ASQR model. The F1 score is used to evaluate the similarity between the predicted answer and the golden answer.

4.3 Quality of Question Reformulation

We first evaluate the accuracy of our question reformulation model in Section 2 leveraging the annotation dataset **dataPretrain**.

Compared Models The compared models of our question reformulation model are as follows:

(1) **Generate**: Attention generator model in (Nallapati et al., 2016). In this model, the words are only generated from a fixed vocabulary.

(2) **Ptr-Generate**: Pointer Generator model in (See et al., 2017). In this model, the word can be copied from the input sentence or generated from the vocabulary. Here, we concatenate the conversational history information and the current question as the input sentence.

(3) **Ptr-Net**: Pure pointer-based copy model with an encoder and a decoder, the input of encoder can be the concatenation of question and conversation history, the decoder only copies words from the input sentences.

(4) **Ptr-Copy**: Pointer copy model is our question reformulation model in Section 2. The word can be either copied from the input questions or copied from the input conversational history.

Results Each question in the annotated dataset has its label reformulated by annotators, so the similarity score between question and its label can be used to evaluate the quality of question reformulation model. The metrics of the similarity scores are BLEU-1,2,3,4, EM (the exact match score), ROUGE_L and F1 scores. The current question may be strongly related to the previous several questions/answers but not all questions/answers history occasionally since topic switching may occur during a conversation. At the same time, sentences containing all history information are longer, which may be not conducive to learning

Model	F1	HEQ-Q	HEQ-D
Pretrained InferSent	20.8	10.0	0.0
Logistic regression	33.9	22.2	0.2
BiDAF++(no-ctx)	50.2	43.3	2.2
ASQR	**53.7**	**48.1**	**2.9**
human	80.8	100	100

Table 4: F1, HEQ-Q and HEQ-D scores on the test dataset of QuAC dataset.

Model	F1	HEQ-Q	HEQ-D
Bert	51.6	46.6	2.9
Ptr-Copy-Bert(4-qa)	52.5	46.9	2.7
Ptr-Copy-Bert(all-qa)	53.1	47.8	2.9
ASQR	**54.2**	**48.5**	**2.9**

Table 5: Model performance on the validation dataset of QuAC dataset.

key information. To verify the above conjecture, we encode previous N questions/answers as conversational history, $N = \{4, all\}$. The results are listed in Table 3. Several conclusions can be drawn from the results:

(1) The Generate model performs poorly since all words in the annotated questions are from the question Q or the conversational history D.

(2) The inferior effect of the Ptr-Generate and Ptr-Net models over our Ptr-Copy model shows that separately encoding the question Q and the conversational history D are better than concatenating them. Because most words in reformulated questions are copied from Q, only referential and missing information needs to be copied from D.

(3) Our Ptr-Copy model with previous all question/answers history performing well proves that our question reformulation model can identify key information accurately in the case of topic switching and longer sentences.

4.4 Effectiveness of ASQR Model

We validate the reformulated data by our ASQR model are more effective for conversational machine comprehension in all **QuAC** dataset.

Compared Models The compared models of our ASQR model are as follows:

(1) **Pretrained InferSent**: Lexical matching baseline model outputting the sentence in paragraph whose pretrained InferSent representation has the highest cosine for the question.

(2) **Logistic regression**: Logistic regression model trained by Vowpal Wabbit dataset (Langford et al., 2007) with simple matching features, bias features and contextual features.

(3) **BiDAF++(no-ctx)**: Single-turn machine comprehension model based on BiDAF (Seo et al., 2016) with self-attention and contextualized embeddings (Peters et al., 2018).

The above three models are baseline models proposed in (Choi et al., 2018). The following models are used in our model.

(4) **Bert**: The pretrained single-turn machine comprehension model with Bert model and one additional output layer trained by official QuAC data.

(5) **Ptr-Copy-Bert**: Get reformulated QuAC data by Ptr-Copy model in Section 2, and train Bert model with the reformulated QuAC data.

(6) **ASQR**: Our ASQR model, an answer-supervised question reformulation model for conversational machine comprehension with reinforcement learning technology. We use the reformulated data by ASQR model to train the Bert model.

Results It is worth noting that the questions in official QuAC dataset do not have labels. The quality of reformulated questions only can be evaluated by their answers. A model is better if the reformulated questions by this model are more beneficial to get better answers. Therefore, we use the similarity scores between predicted answers from single-turn machine comprehension model and the gold answers as the evaluation parameters. The metrics of similarity scores are F1 and HEQ (Human Equivalence score, HEQ-Q for question, HEQ-D for dialog), wherein HEQ-Q is true when the F1 score of the question is higher than the average human F1 score, and HEQ-D is true when the HEQ-Q score of all the questions in the dialog are true.

Table 4 shows the scores on the test dataset of QuAC dataset compared with some baseline models. Our ASQR model has the best F1 (53.7), HEQ-Q (48.1) and HEQ-D (2.9) scores over the baseline models, indicating that the question reformulation model can be beneficial to conversational machine comprehension.

At the same time, some ablation studies have developed on the validation dataset (Table 5). Compared with the Bert trained with original official QuAC dataset, we observe 2.6-improvement on F1 score. The model Ptr-Copy-Bert(all-qa) with the all question/answers history over the model Ptr-Copy-Bert(4-qa) with the part of conversational history has good performance, which is consistent with the result in Section 4.3. The best performance on F1 and HEQ-Q score of our ASQR model compared with the Ptr-Copy-Bert models prove that our answer-supervised training method is more effective than traditional question label-supervised method. Some examples of reformulation data by ASQR over Ptr-Copy model are mentioned in the supplementary section.

Analysis We should point out that the aim of our paper is to prove the effectiveness of answer-supervised question reformulation model. But only question reformulation cannot reach the best performance for conversational machine comprehension problems, because question turns, scenario transformation, answer lapse, et al. are all important factors. The models in Leaderboard such as FlowQA, BiDAF++ w/2 have considered the above import factors, other models such as TransBERT, BertMT use a large amount of data for other tasks. Therefore, it is unfair to compare our model with those models.

Besides, the feedback mechanism of the ASQR model is not good enough because single-turn machine comprehension model does not give appropriate answers occasionally trained by the original QuAC dataset, which severely limits the performance improvement of ASQR model. Some similar question answering models (Buck et al., 2017; Nogueira and Cho, 2017) get feedback by utilizing sophisticated QA system or Search Engine which do not depend on the distribution of input data, while the existing machine comprehension models are strongly dependent on data's distribution. In the future, we will study how to get correct and appropriate feedback, and combine question reformulation with implicit conversational models to better integrate conversational information.

5 Related Work

Recently, several approaches have been proposed for conversational machine comprehension. BiDAF++ w/ k-ctx (Choi et al., 2018) integrates the conversation history by encoding turn number to the question embedding and previous N answer locations to the context embedding. FlowQA (Huang et al., 2018) provides a FLOW mechanism that encodes the intermediate representation of the previous questions to the context embedding when processing the current question. SDnet (Zhu et al., 2018) prepends previous questions and answers to the current question and leverages the contextual embedding of BERT to obtain an understanding of conversation history. The existing models always integrate the conversational history implicitly and can not understand the history effectively.

It is worth noting that much work has introduced question reformulation models into machine comprehension tasks (Feldman and El-Yaniv, 2019; Das et al., 2019). Many question reformulation models can integrate the conversational history explicitly by making coreference resolution and completion for the current question. Rastogi et al. (Rastogi et al., 2019) prove that can get a better answer when inputting a reformulated question to the single-turn question answering models. Nogueira et al. (Nogueira and Cho, 2017) introduce a query reformulation reinforcement learning system with relevant documents recall as a reward. Buck et al. (Buck et al., 2017) propose an active question answering model with reinforcement learning, and learn to reformulate questions to elicit the best possible answers with an agent that sits between the user and a QA system. However, the above work is still in the preliminary exploratory stage, and there is no work to reformulate questions with feedback from downstream tasks in conversational machine comprehension tasks. How to train the reformulation models with feedback from subsequent functions is still a major challenge.

6 Conclusion

In this paper, we present an answer-supervised question reformulation model for conversational machine comprehension with reinforcement learning technology. We provide a high-quality dataset for question reformulation in conversational machine comprehension. The experimental results on a benchmark dataset prove that our model can be more beneficial to improve the performance of conversational machine comprehension.

Acknowledgements

The work was supported by the National Key R&D Program of China under grant 2018YFB1004700, and National Natural Science Foundation of China (61772122, 61872074). We sincerely thank the anonymous reviewers for their thorough reviewing and valuable suggestions.

References

Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *International Conference on Neural Information Processing Systems*.

Christian Buck, Jannis Bulian, Massimiliano Ciaramita, Wojciech Gajewski, Andrea Gesmundo, Neil Houlsby, and Wei Wang. 2017. Ask the right questions: Active question reformulation with reinforcement learning. *arXiv preprint arXiv:1705.07830*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. *arXiv preprint arXiv:1808.07036*.

Rajarshi Das, Shehzaad Dhuliawala, Manzil Zaheer, and Andrew McCallum. 2019. Multi-step retriever-reader interaction for scalable open-domain question answering. *arXiv preprint arXiv:1905.05733*.

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

Yair Feldman and Ran El-Yaniv. 2019. Multi-hop paragraph retrieval for open-domain question answering. *arXiv preprint arXiv:1906.06606*.

Minghao Hu, Yuxing Peng, Zhen Huang, Nan Yang, Ming Zhou, et al. 2018. Read+ verify: Machine reading comprehension with unanswerable questions. *arXiv preprint arXiv:1808.05759*.

Hsin-Yuan Huang, Eunsol Choi, and Wen-tau Yih. 2018. Flowqa: Grasping flow in history for conversational machine comprehension. *arXiv preprint arXiv:1810.06683*.

Marcin Junczys-Dowmunt and Roman Grundkiewicz. 2017. An exploration of neural sequence-to-sequence architectures for automatic post-editing. *arXiv preprint arXiv:1706.04138*.

Yaser Keneshloo, Tian Shi, Chandan K Reddy, and Naren Ramakrishnan. 2018. Deep reinforcement learning for sequence to sequence models. *arXiv preprint arXiv:1805.09461*.

John Langford, Lihong Li, and Alex Strehl. 2007. Vowpal wabbit online learning project.

Kenton Lee, Luheng He, Mike Lewis, and Luke Zettlemoyer. 2017. End-to-end neural coreference resolution.

Hengrui Liu, Wenge Rong, Libin Shi, Yuanxin Ouyang, and Zhang Xiong. 2018. Question rewrite based dialogue response generation. In *International Conference on Neural Information Processing*, pages 169–180. Springer.

Xiaodong Liu, Yelong Shen, Kevin Duh, and Jianfeng Gao. 2017. Stochastic answer networks for machine reading comprehension. *arXiv preprint arXiv:1712.03556*.

Minh Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. *Computer Science*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence rnns and beyond. *arXiv preprint arXiv:1602.06023*.

Jan Niehues, Eunah Cho, Thanh-Le Ha, and Alex Waibel. 2016. Pre-translation for neural machine translation. *arXiv preprint arXiv:1610.05243*.

Rodrigo Nogueira and Kyunghyun Cho. 2017. Task-oriented query reformulation with reinforcement learning. *arXiv preprint arXiv:1704.04572*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. *arXiv preprint arXiv:1802.05365*.

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

Pushpendre Rastogi, Arpit Gupta, Tongfei Chen, and Lambert Mathias. 2019. Scaling multi-domain dialogue state tracking via query reformulation. *arXiv preprint arXiv:1903.05164*.

Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. 2016. Self-critical sequence training for image captioning.

Stefan Riezler and Yi Liu. 2010. Query rewriting using monolingual statistical machine translation. *Computational Linguistics*, 36(3):569–582.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks.

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

Fu Sun, Linyang Li, Xipeng Qiu, and Yang Liu. 2018. U-net: Machine reading comprehension with unanswerable questions. *arXiv preprint arXiv:1810.06638*.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. In *Advances in Neural Information Processing Systems*, pages 2692–2700.

Wei Wang, Ming Yan, and Chen Wu. 2018. Multi-granularity hierarchical attention fusion networks for reading comprehension and question answering. *arXiv preprint arXiv:1811.11934*.

Ronald J. Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning*, 8(3-4):229–256.

Wenhan Xiong, Thien Hoang, and William Yang Wang. 2018. Deeppath: A reinforcement learning method for knowledge graph reasoning.

Qingyu Yin, Zhang Yu, Weinan Zhang, Ting Liu, and William Yang Wang. 2018. Deep reinforcement learning for chinese zero pronoun resolution.

Qingyu Zhou, Yang Nan, Furu Wei, and Zhou Ming. 2018. Sequential copying networks.

Chenguang Zhu, Michael Zeng, and Xuedong Huang. 2018. Sdnet: Contextualized attention-based deep network for conversational question answering. *arXiv preprint arXiv:1812.03593*.