Answer-guided and Semantic Coherent Question Generation in Open-domain Conversation

Weichao Wang , Shi Feng , Daling Wang , Yifei Zhang
School of Computer Science and Engineering, Northeastern University, Shenyang, China
wangwecha@gmail.com
{fengshi,wangdaling,zhangyifei}@cse.neu.edu.cn

Abstract

Generating intriguing question is a key step towards building human-like open-domain chatbots. Although some recent works have focused on this task, compared with questions raised by humans, significant gaps remain in maintaining semantic coherence with the posts, which may result in generating dull or deviated questions. We observe that the answer has strong semantic coherence to its question and post, which can be used to guide question generation. Thus, we devise two methods to further enhance semantic coherence between post and question under the guidance of answer. First, the coherence score between generated question and answer is used as the reward function in a reinforcement learning framework, to encourage the cases that are consistent with the answer in semantic. Second, we incorporate adversarial training to explicitly control question generation in the direction of question-answer coherence. The extensive experiments show that our two methods outperform state-of-the-art baseline algorithms with large margins in raising semantic coherent questions.

1 Introduction

Neural question generation (NQG) has been extensively studied because of its broad application in question answering (QA) systems (Tang et al., 2018; Duan et al., 2017), reading comprehension (Kim et al., 2019; Sun et al., 2018), and visual question answering (Fan et al., 2018). However, the research of open-domain conversational question generation (CQG) (Wang et al., 2018; Hu et al., 2018) is still in its infancy. Different from traditional NQG task, the goal of CQG is to enhance the interactiveness and persistence of chit-chatting. Raising appropriate questions in the conversational system is essential because a mature system should well interact with users by not only responding posts but also asking questions (Li et al., 2016b). Furthermore, asking intriguing questions can proactively probe the users to go deeper and further into the conversational topic (Yu et al., 2016).

The existing CQG methods usually suffer from **dullness** and **deviation** problem, because the semantic coherence is easily disrupted. Wang et al. (2018) first studied the CQG task with PMI model to predict topic words in the inference, which may incorporate noise and generate deviated questions. Hu et al. (2018) proposed an aspect-based question generation model with an encoder-decoder method that did not consider the diversity and might raise dull questions.

Taking Table 1 as a motivating example, the topic of this conversation is about cooking and dishes. For the initial post about enjoying cooking, the appropriate response is a question asking about the special dishes or making appetizers, and the answer tells the exact dish, i.e. "*beef*". If we ignore the answer information, a dull question (e.g. "*What do you mean?*") or a deviated question (e.g. "*How about going singing?*") may be raised.

Table 1 shows that there exists semantic coherence in the post-question-answer conversational thread. The performance limitations of previous CQG literature partially lie in that they only consider the post to generate the question, but the answer that contains rich conversational semantics is ignored. In reading comprehension task, answer information could help to identify appropriate interrogative words (Kim et al., 2019), or copy appropriate phrase by position-aware attention (Sun et al., 2018). Similarly, answer information is leveraged to established the connection between QA and QG (Tang et al., 2018). Although the answer semantics are very effective in traditional NQG tasks, they have never been utilized for generating questions in conversations. Unlike NQG that

Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, pages 5066–5076,
Hong Kong, China, November 3–7, 2019.

● Semantic coherence ● Dullness ● Deviation

Post
I like cooking more and more.
Question candidates
What are your special dishes ? ✓
Are you good at making appetizers ? ✓
What do you mean? ✗
How about going singing? ✗
Answer
Wow, I am only skillful in cooking beef.

Table 1: A motivating example of CQG task, including the post, question, and answer. In question candidate set, we list two intriguing questions and the other two unappropriate questions. The green words indicate the semantic coherence phenomenon. The blue and red words state the dull and deviated responses respectively. Note that the questions asking about the special dishes or making appetizers are all reasonable, and there is no exact match between question and answer.

the question and answer are manually designed according to the context and its goal is to generate an **accurate** and **exact matching** question, the triples are all **casual** and **non-goal-oriented** conversational utterances in CQG as shown in Table 1.

In this paper, we intend to utilize semantics in answer to guide conversational question generation. To the best of our knowledge, there is no prior work incorporating coherence between questions and answers into the CQG task. It is quite challenging to include question-answer semantic coherence into the objective function of supervised CQG models because the coherence also depends on answer information that does not exist in the inference process, i.e. we do not know the answer to the question in advance in a live chatbot. The unavailable answers during inference is the main difference between traditional NQG and CQG task.

To tackle these challenges, we propose two separate learning frameworks based on reinforcement learning (RL) (Sutton et al., 2000) and generative adversarial network (GAN) (Goodfellow et al., 2014) respectively. For the RL framework, we utilize semantic coherence as an estimation of question quality, with inspiration of the fact that the semantic (topic) is coherent in the same question-answer pair. We propose a GRU-MatchPyramid model that incorporates a bi-directional GRU into MatchPyramid model (Pang et al., 2016) to capture higher level's word semantic. Then we employ the pretrained GRU-MatchPyramid to measure the coherence between a question and its corresponding answer. After that, the coherence model is regarded as the environment in an RL framework for optimization, which will guide the learning process to penalize the dull and deviated questions. In addition, the environment is not required during inference.

For the GAN framework, we incorporate adversarial training to explicitly improve coherence between questions and answers. We jointly train a question generator and a discriminator (i.e. GRU-MatchPyramid model) that measures the semantic coherence between generated questions (as well as ground-truth questions) and answers. Only a well-trained generator is needed in inference.

The conditional variational autoencoders (C-VAE) is effective in capturing the diversity of valid responses (Zhao et al., 2017). Based on the advantage of CVAE, we further incorporate prior knowledge of question type to generate questions with reasonable type. Finally, we integrate CVAE into RL framework and GAN frameworks respectively, and propose two novel CQG models, dubbed as RL-CVAE and A-CVAE. Our contributions are summarized in three folds:

- We are the first to incorporate the answer factor into CQG task. The semantic coherence between questions and answers could guide the model to generate more appropriate questions.
- We propose two novel answer-guided semantic coherent CQG models, i.e. RL-CVAE and A-CVAE. The answer information is exploited with reinforcement learning and adversarial learning respectively.
- We conduct comprehensive experiments on our crawled Reddit conversation dataset to confirm the effectiveness of our proposed model. The experimental results demonstrate that our models consistently outperform strong baseline methods.

2 Related Work

The neural network-based methods have been applied successfully in natural language generation problems, such as text summarization (See et al., 2017; Chen et al., 2019), machine translation (Bahdanau et al., 2015; Vaswani et al., 2017) and question generation. Traditional NQG tasks are explored in reading comprehension (Kim et al.,

2019; Sun et al., 2018), QA systems (Tang et al., 2018; Duan et al., 2017). In such tasks, the answer is known and is part of the input to the generated question. The answer information could help to predict reasonable interrogative words (Kim et al., 2019), or perceive appropriate phrase by modelling the relative distance between the context words and the answer (Sun et al., 2018), or bridge the relevance between QA and QG tasks (Tang et al., 2018).

Furthermore, NQG can also be deployed as chatbot components, such as visual question answering (Fan et al., 2018), task-oriented dialogues (Li et al., 2016b), and open-domain dialogue generation (Wang et al., 2018; Hu et al., 2018). Li et al. (2016b) devised several handcrafted templates, which is not applicable to open-domain conversational systems. Wang et al. (2018) first studied on open-domain CQG, and they devised soft and hard typed decoders by capturing different roles of different word types, and used PMI model to predict topic words. Hu et al. (2018) proposed an aspect-based question generation problem, and used an encoder-decoder model to generate aspect-specific questions.

The RL has been widely applied to conversational generation (Zhou and Wang, 2018; Li et al., 2016c; Heeman et al., 2012; Lemon et al., 2014), and the GAN has been successfully applied to text generation (Li et al., 2018; Yu et al., 2017). However, these methods did not explore the answer information. Inspired by the important role of answer information in traditional NQG tasks, we intend to use RL and GAN to solve the open-domain CQG problem. To the best of our knowledge, we are the first to incorporate answer and conversational content into the both RL and GAN frameworks for this task.

3 Proposed Models

3.1 Problem Formalization

Following the conversation question generation paradigm (Wang et al., 2018), our method has a similar task setting where a question is generated based on a post utterance. For the particularity of our task, we further leverage the question type to control and interpret the generated question, and take advantage of the semantic coherence guided by answer information to improve the quality of generated questions.

The conversational content (U, q, l, a), consists of post utterance U, response utterance (i.e. question) q with its question type l, and answer utterance a to question q. In training, we aim to estimate the probability $p(q, l|U)$ utilizing C-VAE, and use reinforcement learning and adversarial learning to explore the answer information a with GRU-MatchPyramid measuring the coherence. In inference, given a new post utterance U, we can generate an appropriate question q' and question type l' according to the generation probability. Note that the answer utterance a is only used in training process, and is unknown in inference process.

3.2 Conditional Variational Autoencoder

The conditional variational autoencoder (C-VAE) (Sohn et al., 2015) is an extention of sequence-to-sequence model, and proves to be very effective in promoting the diversity in conversation generation (Serban et al., 2017). Besides, prior linguistic features (e.g. dialogue acts (Zhao et al., 2017) or sentiment (Shen et al., 2017)) have been incorporated into CVAE for controlling and interpreting dialogue generation. Inspired by previous works, we apply CVAE model to generate diverse and meaningful questions for CQG task. Furthermore, question type can provide semantic hints that enhance the semantic relevance for generating controllable questions. So we incorporate question type into CVAE for controlling question generation with reasonable type.

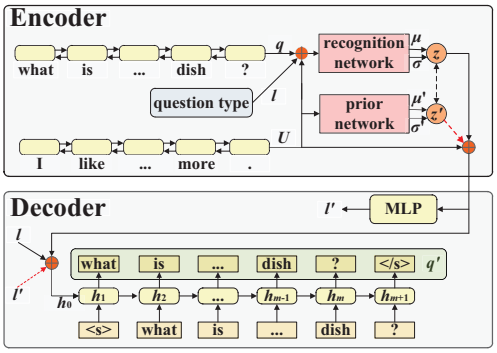

Figure 1: The CVAE model incorporated with question type embedding. $\bigoplus$ denotes the vector concatenation operation. In training process, approximated poster latent variable z obtained from recognition network, together with true question type l and post utterance U are concatenated for decoder process. At the same time, red dashed arrows refer to inference process, where we replace z with prior latent variable z' obtained from prior network, and replace l with predicted question type l' in decoder process.

The CVAE model introduces a latent variable z to capture the distribution over responses, so the generation probability is defined as $p(q, z, l|U) = p(q|z, U, l)p(l|z, U)p(z|U)$. As shown in Figure 1, the encoder neural network is formulated as the bi-directional GRU model, which is used to encode post U and question q. The question type l is represented by a real-valued, low dimensional vector which is learnt through training. Note that the reason why we use the actual question type l during training is to generate appropriate question consistent with the ground-truth in content and question type simultaneously. By assuming z follows multivariate Gaussian distribution, the prior network parameterized by θ_P is introduced to approximate $p_P(z'|U) \sim \mathcal{N}(\mu', \sigma'^2\mathbf{I})$. The recognition network parameterized by θ_R is introduced to approximate $p_R(z|U, q, l) \sim \mathcal{N}(\mu, \sigma^2\mathbf{I})$. The latent variables are sampled by reparameterization trick. The question type prediction network parameterized by θ_L is introduced to approximate $p_L(l|U, z)$. The decoder neural network parameterized by θ_D is formulated as a 1-layer GRU model, which is used to approximate $p_D(q|z, U, l)$. During training, z obtained by recognition network, together with l and U are used to approximate $p_D(q|z, U, l)$. The response decoder then gets sematic representation h_1,...,h_{m+1} with predicting the words of response. During inference, z' obtained by prior network, together with predicted question type l' and U are regarded as input of decoder process, which corresponds to red dashed arrows in Figure 1.

Finally, the CVAE model incorporated with question type is trained by maximizing both the variational lower bound and question type prediction accuracy:

$$\begin{aligned}\mathcal{L}_{CVAE} = &-KL(p_R(z|U, q, l)||p_P(z|U)) \\ &+E_{p_R(z|U,q,l)}logp_D(q|z, U, l) \\ &+E_{p_R(z|U,q,l)}logp_L(l|z, U)\end{aligned} \quad (1)$$

We use KL annealing (Bowman et al., 2016) and bag-of-word loss (Zhao et al., 2017) to avoid the vanishing latent variable problem.

3.3 GRU-MatchPyramid Model

In this paper, we formulate measuring the semantic coherence between questions and answers as a text matching task.

The MatchPyramid (Pang et al., 2016) considers both semantic representation and semantic interactions in text matching. However, for large-scale conversational text, it is more essential to capture the word's higher level semantic rather than superficial meaning, as quite a part of sentence pairs in conversation do not have the same keywords but only contain semantic-related words (e.g. "*dish*" in ground truth question and "*cooking*", "*beaf*" in answer of Table 1). So we introduce bi-directional GRU into MatchPyramid model's word level to capture higher level's word semantic, and propose a new GRU-MatchPyramid model.

For question q's i-th word w_i with embedding e_{w_i}, and answer a's j-th word v_j with embedding e_{v_j}, the matching matrix M is defined as:

$$M_{ij} = BiGRU(e_{w_i}) \otimes BiGRU(e_{v_j}) \quad (2)$$

where M_{ij} is semantic coherence degree between word w_i and word v_j, $BiGRU(e_{w_i})$ and $BiGRU(e_{v_j})$ are words' high-level semantic representation via bi-directional GRU, and $\otimes$ is dot product operation. Then a one-layer convolutional neural network is used to extract the matching pattern based on M, and the features matrix is obtained. Afterwards, a max-over-time pooling operation (Collobert et al., 2011) over the features matrix obtained by convolutional operation is used to capture the most important information M'. We flatten M' and use a MLP model to generate the semantic coherence feature f. The semantic feature M', coherence feature f and coherence score s are defined as:

$$M' = CNN(M) \quad (3)$$

$$f = \varphi(W_f \cdot flatten(M') + b_f) \quad (4)$$

$$s = \varphi(W_s f + b_s) \quad (5)$$

where φ is the rectified linear units activation function, i.e. ReLU, W_f, b_f, W_s, b_s are trainable parameters.

3.4 RL-CVAE

Inspired by reinforcement learning in controlling characteristics of generated responses (e.g. grammatical coherence (Li et al., 2016c), or sentiment expression (Zhou and Wang, 2018)), we conjecture that promoting the question-answer semantic coherence via reinforcement learning can help generate appropriate questions. The supervised learning cannot be applied to calculate question-answer semantic coherence, as it suffers from the non-differentiable problem caused by the sampling-based output decoding procedure.

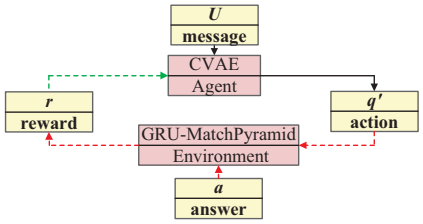

Figure 2: The CVAE with reinforcement learning (RL-CVAE). We formulate the CVAE model as an agent and pretrained GRU-MatchPyramid model as an environment in an RL framework. Solid arrows present the process of generating action q' (i.e. generated question) on the condition of state U (i.e. post). Red dashed arrows refer to agent's interaction process with the environment, where reward r is obtained by using action q' and answer a. Green dashed arrow refer to policy gradient optimization process with reward r.

To explicitly promote the semantic coherence between questions and answers, we propose a RL-CVAE model, as shown in Figure 2. Specifically, RL-CVAE is initialized by the pretrained CVAE model that acts as an agent in reinforcement learning. The action q' is the generated question, and the state is denoted by the post U and will affect the action. The policy of agent takes the form of CVAE, and it will output a probability distribution over actions conditioned on given states U. The pretrained GRU-MatchPyramid model acts as an environment, which considers answer a and question q' as input, and calculates the reward function to output reward value r. The reward value r (i.e. semantic coherence score) can be interpreted as quality estimation of q' so that it can guide the learning process to reward the cases that have strong coherence to the answer, and to penalize the cases that have little coherence to the answer (i.e. the dull and deviated cases).

We utilize pretrained GRU-MatchPyramid model as the measuring method to evaluate semantic coherence between q' and a. When training GRU-MatchPyramid model, we use negative sampling method to construct the training data, which has been successfully applied in retrieval-based chatbot (Zhou et al., 2018; Zhang et al., 2018) and text matching task (Liu et al., 2016). Specifically, for each question (ground-truth) and answer pair (q, a) in the conversation generation training data, we randomly sample five negative questions q^-, chosen from other posts' ground-truth responses. The training objective is that the score of (q, a) should be larger than (q^-, a) with at least $\triangle$ threshold. The loss function is defined as:

$$\mathcal{L}_{coh} = max(\triangle - s(q, a) + s(q^-, a), 0) \quad (6)$$

where s is semantic coherence score between question and answer as defined in Equation 5.

After obtaining a generated question q', we get coherence score ${r_0}^+ = s(q', a)$ between generated question and answer. However, ${r_0}^+$ has a wide range of value that it could not be used as reward value in reinforcement learning directly. Similar with GRU-MatchPyramid's training process, we create a coherence score set R^- with negative examples, and use min-max scaling to normalize the coherence score ${r_0}^+$. Specifically, we randomly select n negative questions ${q_1}^-, {q_2}^-, ... {q_n}^-$ from other posts's ground-truth responses, and then calculate their coherence scores with a via Equation 5, i.e. $R^- = ({r_1}^-, {r_2}^-, ..., {r_n}^-)$. The final coherence score $r(q', a)$ is then defined as follows:

$$r = \frac{{r_0}^+ - min(R^-, {r_0}^+)}{max(R^-, {r_0}^+) - min(R^-, {r_0}^+)} \quad (7)$$

where $min(R^-, {r_0}^+)$ and $max(R^-, {r_0}^+)$ are the minimum value and maximum value of set $(R^-, {r_0}^+)$. We use the policy gradient methods (Sutton et al., 2000) for optimization with the expected reward defined as:

$$J(\theta) = \mathbb{E}_{p(q,z,l|U)}[r(q, a)] \quad (8)$$

The gradient is estimated as:

$$\nabla J(\theta) = r(q, a) \nabla log p(q, z, l|U) \quad (9)$$

3.5 A-CVAE

The GAN (Goodfellow et al., 2014) is effective in controlling attributes of generated text (e.g. sentiment (Logeswaran et al., 2018) or thematic (Li et al., 2018)). Based on our assumption that C-QG task can benefit from question-answer semantic coherence, we propose an A-CVAE model with a question generator and a question-answer coherence discriminator to further improve the quality of generated questions.

In Figure 3, the generator (i.e. CVAE model) is used to generate questions q', and the discriminator (i.e. GRU-MatchPyramid model) is incorporated to learn the matching score between generated question q' (as well as ground-truth question q)

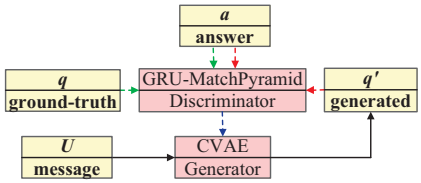

Figure 3: The CVAE with adversarial learning (A-CVAE). We formulate the CVAE as a generator and GRU-MatchPyramid as a discriminator using GAN. Solid arrows present the generation process of question q' conditions on post utterance U. Red dashed arrows refer to the adversarial learning process of negative instance (i.e. q' and answer a), and green dashed arrows refer to the positive instance (i.e. ground-truth question q and a). Blue dashed arrow refer to the back-propagation of the discriminator to the generator.

and answer a. In the discriminator, we treat q' as the negative sample incoherent to answer a, and treat q as the positive sample, which on the contrary is coherent to a. Then the loss from this discriminator is back-propagated to the decoder process of the generator to improve the coherence.

Inspired by (Li et al., 2018), to better perform gradient calculation in adversarial process, the decoder's state sequence $(h_1, h_2, ..., h_m)$ shown in Figure 1 is used to represent generated question q'. To align dimension between q' and a, we conduct nonlinear transformation on state h_i:

$$h'_i = \varphi(W_d h_i + b_d) \tag{10}$$

After transformation, the q' is then denoted as $q' = (h'_1, h'_2, ..., h'_m)$. According to Equation 5, we get the semantic coherence score s_g between q' and a (i.e. $s_g = s(q', a)$), and score s_t between the ground-truth question q and a (i.e. $s_t = s(q, a)$).

Following the routine of improved wasserstein generative adversarial networks, i.e. WGAN-PG (Gulrajani et al., 2017), the discriminator is trained to measure the semantic coherence degree between generated (as well as ground truth) questions and answers, according to the semantic coherence scores s_g and s_t, with minimizing the objective function:

$$\mathcal{L}_D = s_g - s_t \tag{11}$$

Note that the discriminator which utilizes the answer is only applied during the training process, because the answer information is unavailable during inference.

The overall objective of generator is to minimize

$$\mathcal{L}_G = -\mathcal{L}_{CVAE} - \lambda s_g \tag{12}$$

with respect to parameters of the CVAE, where $\mathcal{L}_{CVAE}$ is defined in Equation 1. λ is a balancing parameter. Intuitively, during training the generator is trying to generate answer-coherent questions that would confuse the discriminator as to these questions' origin (i.e. ground-truth or generated) by maximizing $\mathcal{L}_D$, so that the discriminator is forced to become a strong semantic coherence evaluator to distinguish these questions' origin by minimizing $\mathcal{L}_D$, until both the generator and discriminator can not improve any more. In this way, the adversarial learning can push generator to generate questions indistinguishable from ground-truth.

4 Experiment

4.1 Experiment Setup

We extract our experiment corpus on Reddit[1], and the comment utterances are crawled for our experiment evaluation[2]. Inspired by (Fan et al., 2018), we categorize questions into 9 types, i.e. "what", "when", "where","who","why" "how","can (could)","do (did, does)","is (am, are, was, were)" according to interrogative words that are the most significant features to distinguish question types. We filter out the crawled data whose questions are not included in these 9 types, because these data occupy a small proportion in the crawled conversation dataset, and it is difficult to learn satisfying question type representation. Ultimately, we collected 1,164,345 pieces of data, and each of which has a post, a question, and an answer. We randomly selected 30,000 triples for validation and another 30,000 triples for testing. The average number of words in post/response/answer is 18.84/19.03/19.30 respectively. We choose top 40,000 frequent words as the vocabulary.

We use the 200d Glove embedding (Pennington et al., 2014) pretrained on Wikipedia 2014 dataset as word embedding. In CVAE, the size of encoder state is set to 300, and the size of decoder state is set to 400. We set the size of latent Gaussian variable to 200, the size of question type to 100, and the size of mini-batch to 30. The prior network and question type prediction network have a hidden layer whose dimension is 400. All weights are initialized by the xavier method (Glorot and

[1]http://www.reddit.com

[2]The dataset is available at https://drive.google.com /drive/folders/1wNG30YPHiMc_ZNyE3BH5wa1uVtR8l1pG

Bengio, 2010). The dataset is tokenized using the NLTK tokenizer. We optimize our model using Adam (Kingma and Ba, 2014) with learning state of 0.001 and gradient clipping at 5. We use KL-annealing strategy with the temperature varying from 0 to 1 and increased by 1/60k after each iteration of batch update.

For the GRU-MatchPyramid model, the bi-directional GRU model for encoding question and answer has a hidden state of 300 in size. In the convolutional operation, the kernel size is set to (3,3), with the stride size to 2.

In RL-CVAE, we set threshold $\triangle$ to 1 in GRU-MatchPyramid's training, and set number n to 100 in min-max scaling. The pretrained CVAE model is obtained by bag-of-words loss along with KL annealing to 0.5 after training 30k batches. Then we use policy loss to obtain the final model with KL weight kept at 0.5. In A-CVAE, we set balancing parameter λ in generator's loss to 0.1, and train the discriminator one time for every five times training of generator.

4.2 Baselines

Seq2Seq: A simple sequence-to-sequence model with attention mechanism.
CVAE: Conditional variational autoencoders with an auxiliary bag-of-words loss is used to generate diverse responses (Zhao et al., 2017).
STD&HTD: The STD uses soft typed decoder by estimating words' distribution over word types, and HTD uses hard typed decoder by specifying the type of each word explicitly (Wang et al., 2018). Because these two methods depend on PMI, we collected about 100,000 post-response pairs from Reddit to estimate the probabilities in PMI. The HTD is the state-of-the-art model for C-QG task.

4.3 Question Generation Evaluation

Perplexity: The perplexity (PPL) (Vinyals and Le, 2015) could evaluate whether generated responses satisfy the grammatical rules, and lower value reflects better fluency.
RUBER: RUBER (Referenced metric and Unreferenced metric Blended Evaluation Routine) (Tao et al., 2018) has shown a high correlation with human annotation in open-domain conversation response evaluation. RUBER measures the similarity between generated responses and ground-truth responses as referenced metric, and measures the relatedness as unreferenced metric. The similarity is calculated via cosine similarity, and the relatedness is obtained by a neural network pretrained via a utterance matching method. Then the geometric averaging (RubG) and arithmetic averaging (RubA) are obtained via these two metrics, and higher value reflects better semantic coherence.
Distinct2 (Li et al., 2016a): We count numbers of distinct bigrams in the generated responses. The ratio (Dist2) is obtained by dividing the counted number by the total number of generated bigrams, and higher value reflects better diversity.

Question Generation Evaluation				
Models	RubA	RubG	Dist2	PPL
Seq2Seq	0.614	0.574	0.008	63.02
CVAE	0.682	0.649	0.112	20.39
STD	0.658	0.613	0.010	28.75
HTD	0.689	0.654	0.114	26.02
CVAE (qt)	0.688	0.652	0.114	20.03
A-CVAE	0.715	0.661	0.123	19.51
RL-CVAE	**0.720**	**0.668**	**0.185**	**16.93**

Table 2: Evaluation result of question generation.

As shown in Table 2, CVAE(qt) incorporates question type information, and slightly improves the performance compared with CVAE, because it could help generate questions with reasonable type. Furthermore, answer information could further improve the overall performance. First, RL-CVAE and A-CVAE perform fairly well with lower perplexities, because they could generate fluent questions. Second, they can obtain higher RubG and RubA scores, showing that taking advantage of coherence between question and answer could further enhance semantic coherence between post and question. Third, they obtain higher distinct values, because answer information provides extra clues to guide question generation.

Our RL-CVAE performs better than A-CVAE. This is because the decoder's hidden states are utilized to represent generated question word tokens in A-CVAE, which is asymmetric to the ground-truth question in adversarial process. Instead, the question sentence is observed in RL-CVAE after the model arrives at the end of a sequence, and it is then used to calculate coherence degree directly.

4.4 Effect of GRU-MatchPyramid

We use mean average precision (MAP) to evaluate the performance of GRU-MatchPyramid model. MAP is the metric to measure the accuracy of tex-

Post	Generated Question	Answer
that's definitely a third party seller at amazon and not amazon .	Seq2Seq: is it a good thing? CVAE: and how is this an official app? STD: how do you like amazon? HTD: are you heavily subscribed to amazon? CVAE(qt): can you get the link? i'm just using the amazon box. **RL-CVAE: what's the difference between a third party seller and amazon ?** **A-CVAE: does the third party seller ship amazon fulfillment?**	it's still third seller so amazon usually ships the item as it was shipped to them but in a box. i would buy directly from amazon.
i don't know why so many people don't realize the duke o death is only free for returning players.	Seq2Seq: what's your point? CVAE: are you just about having fun? STD: what's the difference? HTD: are you returning player? CVAE(qt): are there any members in your opinion? **RL-CVAE: how much is it for normal players?** **A-CVAE: how to realize that? i'm on the list of returning players.**	it's 660k for normal players. and in the newswire they said all the returning player stuff would be free for returning players.

Table 4: Two cases comparison among different models.

Models	MP	GRU-MatchPyramid
MAP	0.793	**0.826**

Table 3: Evaluation result of GRU-MatchPyramid.

t matching, and it is the average of the maximum precisions at different recall values. As shown in Table 3, GRU-MatchPyramid model performs better than MatchPyramid (dubbed as MP) model in large scale conversational data, because of capturing higher level semantic of word. In this way, we get a better reward function in RL-CVAE, and a better discriminator in A-CVAE.

4.5 Question Answer Coherence Evaluation

To evaluate the degree of semantic coherence between generated questions and answers, we apply two evaluation methods: (1) cosine similarity: inspired by sentence-level similarity evaluation (Li et al., 2018), we use embedding-based metrics to calculate cosine similarity, and higher similarity implies better topic similarity. (2) matching score: we use pretrained GRU-MatchPyramid model to calculate the semantic coherence, and higher scores imply better semantic coherence.

As shown in Table 5, our proposed models could generate questions coherent to answers in inference process. The coherence of question and answer improved overall performance shown in Table 2 by a large margin.

4.6 Human Evaluation

We randomly sampled 500 cases and recruited three graduate students as human annotators. To

Models	cosine	matching score
Seq2Seq	0.494	5.304
CVAE	0.591	8.050
STD	0.540	6.882
HTD	0.593	8.063
CVAE (qt)	0.594	8.053
A-CVAE	0.612	8.420
RL-CVAE	**0.617**	**8.512**

Table 5: Evaluation result of question-answer semantic coherence.

Models	A	S	W	Sum
seq2seq	0.486	0.208	0.196	0.890
CVAE	0.458	0.484	0.408	1.350
STD	0.504	0.322	0.272	1.098
HTD	0.528	0.486	0.406	1.420
CVAE(qt)	0.462	0.508	0.468	1.438
A-CVAE	0.540	0.578	0.514	1.632
RL-CVAE	**0.542**	**0.602**	**0.526**	**1.670**

Table 6: Results of human evalution based on criteria of appropriateness (A), semantic coherence (S) and willingness to answer (W).

each annotator, we showed the same post utterance of a test example with all seven questions generated by different models. The seven questions were presented in random order, and the annotators were asked to evaluate whether each question satisfies criteria defined as: (1) **Appropriateness**: measures whether a question is reasonable in logic and grammar. (2) **Semantic coherence**: measures whether the question are coherent to the

given post. Incoherent questions include dull and deviated cases. (3) **Willingness to answer**: measures whether a user is willing to answer the question. This criterion is to justify how likely the generated questions can elicit further interactions.

We calculate the ratio of each criterion satisfied, and regard the sum as the evaluation result. The results are calculated by averaging the annotations from the three judges. We calculated the Fleiss' kappa (Fleiss, 1971) to measure inter-annotator agreement. Fleiss' kappa for **Appropriateness**, **Semantic coherence** and **Willingness to answer** is 0.486, 0.421 and 0.506 respectively, indicating "Moderate agreement" for all three criteria. As shown in Table 6, RL-CVAE and A-CVAE are consistently in line with the human perspective, especially in semantic coherence and willing to answer criteria. It indicates that with utilizing question-answer semantic coherence, the generated questions are more coherent to posts, and our models could help to drive the conversation to go further proactively. Compared with the state-of-the-art model HTD, our proposed models could effectively alleviate the dullness and deviation issues.

4.7 Case Study

As shown in Table 4, we list two cases from test corpus to compare different methods. The topics about post and answer in the two cases are about "*third-party seller of amazon*" and "*returning players*" respectively. The A-CVAE and RL-CVAE model could generate appropriate questions coherent to both posts and answers semantically, showing that answer information of training process helps to control and guide the question generation obviously. However, without exploring answer information, the baseline models may only generate dull questions (e.g. Seq2Seq), or deviated questions (e.g. CVAE, STD, and HTD). Furthermore, based on the given post and generated questions, we are more willing to answer the questions generated by our methods.

5 Conclusion

In this paper, we propose two novel models A-CVAE and RL-CVAE, which can generate semantic coherent questions under the guidance of answers in open-domain conversational system. For A-CVAE model, we use adversarial training to explicitly control question generation in the direction of question-answer coherence. For RL-CVAE, the coherence between questions and answers is leveraged as the reward function in an RL framework. Furthermore, our work of utilizing RL and GAN with answer information is general, and it can be easily extended to existing work.

Acknowledgements

The work was supported by the National Key R&D Program of China under grant 2018YFB1004700, and National Natural Science Foundation of China (61872074, 61772122).

References

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

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*.

Yangbin Chen, Yun Ma, Xudong Mao, and Qing Li. 2019. Multi-task learning for abstractive and extractive summarization. *Data Science and Engineering*, 4(1):14–23.

Ronan Collobert, Jason Weston, Léon Bottou, Michael Karlen, Koray Kavukcuoglu, and Pavel Kuksa. 2011. Natural language processing (almost) from scratch. *Journal of Machine Learning Research*, 12(Aug):2493–2537.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874.

Zhihao Fan, Zhongyu Wei, Piji Li, Yanyan Lan, and Xuanjing Huang. 2018. A question type driven framework to diversify visual question generation. In *IJCAI*, pages 4048–4054.

Joseph L Fleiss. 1971. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the thirteenth international conference on artificial intelligence and statistics*, pages 249–256.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron

Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680.

Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. 2017. Improved training of wasserstein gans. In *Advances in Neural Information Processing Systems*, pages 5767–5777.

Peter A Heeman, Jordan Fryer, Rebecca Lunsford, Andrew Rueckert, and Ethan Selfridge. 2012. Using reinforcement learning for dialogue management policies: Towards understanding mdp violations and convergence. In *Thirteenth Annual Conference of the International Speech Communication Association*.

Wenpeng Hu, Bing Liu, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2018. Aspect-based question generation. In *ICLR Workshop*.

Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and Kyomin Jung. 2019. Improving neural question generation using answer separation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6602–6609.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Oliver Lemon, Srinivasan Chandrasekaran Janarthanam, and Verena Rieser. 2014. Reinforcement learning approaches to natural language generation in interactive systems. In *Natural Language Generation in Interactive Systems*, pages 151–175. Cambridge University Press.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. *Computer Science*.

Jiwei Li, Alexander H Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2016b. Learning through dialogue interactions by asking questions. *arXiv preprint arXiv:1612.04936*.

Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. 2016c. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*.

Juntao Li, Yan Song, Haisong Zhang, Dongmin Chen, Shuming Shi, Dongyan Zhao, and Rui Yan. 2018. Generating classical chinese poems via conditional variational autoencoder and adversarial training. In *EMNLP*, pages 3890–3900.

Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xuanjing Huang. 2016. Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1034–1043.

Lajanugen Logeswaran, Honglak Lee, and Samy Bengio. 2018. Content preserving text generation with attribute controls. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 5103–5113. Curran Associates, Inc.

Liang Pang, Yanyan Lan, Jiafeng Guo, Jun Xu, Shengxian Wan, and Xueqi Cheng. 2016. Text matching as image recognition. In *AAAI*, pages 2793–2799.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. Glove: Global vectors for word representation. In *EMNLP*, pages 1532–1543.

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.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*.

Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017. A conditional variational framework for dialog generation. *arXiv preprint arXiv:1705.00316*.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In *Advances in Neural Information Processing Systems*, pages 3483–3491.

Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *EMNLP*, pages 3930–3939.

Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in neural information processing systems*, pages 1057–1063.

Duyu Tang, Nan Duan, Zhao Yan, Zhirui Zhang, Yibo Sun, Shujie Liu, Yuanhua Lv, and Ming Zhou. 2018. Learning to collaborate for question answering and asking. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, volume 1, pages 1564–1574.

Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. Ruber: An unsupervised method for automatic evaluation of open-domain dialog systems. In *AAAI*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.

Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. 2018. Learning to ask questions in open-domain conversational systems with typed decoders. In *ACL*.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Zhou Yu, Ziyu Xu, Alan W Black, and Alexander Rudnicky. 2016. Strategy and policy learning for non-task-oriented conversational systems. In *Proceedings of the 17th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pages 404–412.

Zhuosheng Zhang, Jiangtong Li, Pengfei Zhu, Hai Zhao, and Gongshen Liu. 2018. Modeling multi-turn conversation with deep utterance aggregation. *arXiv preprint arXiv:1806.09102*.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*, pages 654–664. Association for Computational Linguistics.

Xianda Zhou and William Yang Wang. 2018. Mojitalk: Generating emotional responses at scale. In *ACL*, pages 1128–1137.

Xiangyang Zhou, Lu Li, Daxiang Dong, Yi Liu, Ying Chen, Wayne Xin Zhao, Dianhai Yu, and Hua Wu. 2018. Multi-turn response selection for chatbots with deep attention matching network. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1118–1127.