2019 IEEE International Conference on Data Mining (ICDM)

An Efficient Policy Gradient Method for Conditional Dialogue Generation

Lei Cai
School of Electrical Engineering and Computer Science
Washington State University
Pullman, WA USA
lei.cai@wsu.edu

Shuiwang Ji
Department of Computer Science & Engineering
Texas A&M University
College Station, TX USA
sji@tamu.edu

***Abstract*—Encoder-decoder models have been commonly used in dialogue generation tasks. However, they tend to generate dull and generic response utterances. To tackle this problem, we consider dialogue generation as a conditional generation problem. For a given context history, our model can generate different response utterances with desirable dialog acts. Our model follows the SeqGAN framework, where the generator takes context history and dialog act as inputs and generates corresponding response utterances. The discriminator computes rewards by considering the quality of entire utterance and dialog act. Our model is trained by a policy gradient approach. To overcome the bottleneck of excessive time complexity incurred by the Monte Carlo search for training, we propose a local discriminator network to compute the individual reward in one forward propagation, thereby dramatically accelerating the training procedure. Experimental results demonstrate that our proposed method can achieve comparative performance with Monte Carlo search, while reducing the training time dramatically.**

***Keywords*-deep learning, conditional dialogue generation, policy gradient, local discriminator network**

I. Introduction

Dialogue generation is of great importance in natural language processing research and applications. Dialogue generation models take context history as input and generate response utterances for the input. Usually, there are multiple reasonable response utterances corresponding to a given input as illustrated in Figure 1. Thus, dialogue generation models need to provide diverse and informative responses. With the development of deep learning, many deep models have been developed to tackle dialogue generation tasks [1]–[5]. These methods commonly use the encoder-decoder architecture [6] for open-domain dialogue generation. The encoder-decoder model represents inputs as a hidden state, which is then used to initialize the decoder and generate response utterances. Since the encoder-decoder network is optimized by maximizing the likelihood, it may generate dull and generic responses.

Context:

You know you're told if you find them guilty then this is the choices that they have to make it's either life or death you know.

Dialogue Act	Response Utterance
statement	It seems like in some of these cases I don't know that you could really have an impartial.
agree/accept	Yeah, I think that's true.
wh-question	What do you mean?
acknowledge	Right.

Figure 1. Illustration of the scenario where multiple reasonable response utterances exist for a given context.

In this work, we propose to formulate this task as a conditional dialogue generation problem, where the generated response utterance depends on both the context history and the dialog act. In dialogue systems, each utterance can be labeled with a dialog act such as "statement-opinion", "agree", "apology", "open-question", etc. If the same dialogue context history and different dialog acts are given, the model should generate different response utterances corresponding to the given dialog act. To achieve this goal, we combine the encoder representation with the dialog act to initialize the decoder network that generates the response utterances. However, the encoder-decoder network trained by maximizing the likelihood generates dull and generic response utterances. To address this limitation, we propose to apply the SeqGAN [7] framework for this task. Generative adversarial networks (GANs) were proposed to capture data distributions and generate samples. GANs can only generate continuous data, and it is difficult to use GANs to generate discrete text tokens. In order to apply GANs to generate text data, SeqGAN considers the text generation procedure as a reinforcement learning problem [8]. In SeqGAN, the generator corresponds to the policy network, and the discriminator is used to compute the reward for generated response utterances. The SeqGAN is trained by the policy gradient method [9]. It was shown that SeqGAN can achieve better performance than the encoder-decoder network. To assign each action a reward value, SeqGAN employs Monte Carlo search in the training procedure. To compute the reward of each token, SeqGAN needs to roll out the policy several times. Therefore, the training of SeqGAN is very time-consuming.

2374-8486/19/$31.00 ©2019 IEEE
DOI 10.1109/ICDM.2019.00013

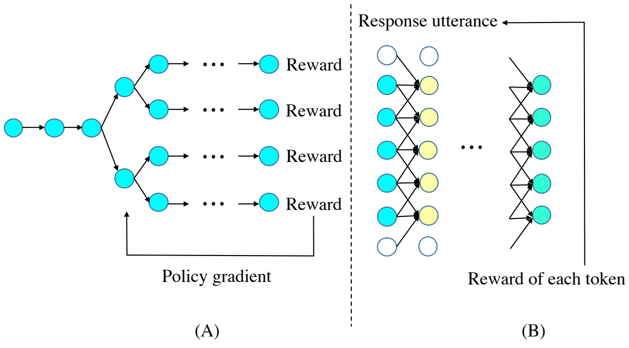

Figure 2. Illustration of reward computation methods by Monte Carlo search (A) and our proposed local discriminator network (B).

In this work, we propose a local discriminator network to compute the reward for each token in the generated response utterances and apply it to the SeqGAN framework for conditional dialogue generation tasks. Instead of generating the response utterance for multiple times by the rollout approach, our proposed method can compute the reward in one forward propagation, thereby dramatically accelerating training as illustrated in Figure 2. We evaluate our proposed method on the Switchboard dataset and Daily Dialog dataset. Experimental results show that our proposed method can achieve comparative performance with the Monte Carlo search method, while dramatically reducing the training time.

Our contributions can be summarized as follows:

1) We propose a conditional dialogue generation model to generate diverse response utterances based on given dialog acts.
2) We propose to employ a discriminator network, incorporating global, local, and dialog act information, to encourage the generator to produce reasonable responses.
3) We propose to compute the reward for each token using a local discriminator network based on CNN, instead of using Monte Carlo search as in prior methods. Our proposed method can compute the reward of each token in one forward pass, thereby reducing training time dramatically while achieving competitive performance.

II. Related Work

The encoder-decoder models [6], [10] have been successfully applied to many natural language generation tasks such as machine translation [11] and document summarization [12]. For dialogue generation tasks, encoder-decoder models capture the relationship between context history and response utterances using recurrent neural networks [13]. The encoder-decoder models are trained end-to-end by maximizing the likelihood. However, these models usually generate dull and generic responses.

Li *et al.* [2] claim that maximizing the likelihood cannot approximate the real-world goal of chatbot development. They consider dialogue generation tasks as a reinforcement learning problem. The encoder-decoder network is first trained by maximizing the likelihood and subsequently improved by maximizing the reward function. The reward function is designed to tackle the generic response problem by considering ease of answering, information flow, and semantic coherence. The key idea of this method is to develop a reward function to evaluate the generated response utterances and penalize dull and generic responses. The manually designed reward function is still limited for a dialogue generation system. Therefore, some researchers propose to employ GANs frameworks and use the discriminator networks as a reward function to evaluate the response utterances.

In [14], GANs are proposed to tackle image generation problems and have achieved impressive performance. The GANs framework consists of a generator network and a discriminator network. The generator network generates images from latent representations. The discriminator is used to distinguish real and generated images. GANs have achieved great success on computer vision tasks [15]–[18]. Since text tokens are discrete, it is difficult to apply GANs on natural language processing tasks.

In [7], SeqGAN is proposed to tackle the non-differentiable problem by considering dialogue generation tasks as a reinforcement learning problem. In the SeqGAN, the generation of each token is considered as an action, and the reward of action is evaluated by a discriminator network using the Monte Carlo search approach. The SeqGAN is trained by a policy gradient method to encourage the generator to produce better outputs. Inspired by the Turing test, Li *et al.* [3] propose to apply GANs on dialogue generation tasks. They train the discriminator network to distinguish human generated responses and machine generated responses. In their model, an encoder-decoder network is trained as the generator to fool the discriminator and produce human-like response utterances. They employ the teacher-forcing method to make the training procedure stable. Yang *et al.* [19] use a convolutional neural network as the discriminator and achieved better performance than that of recurrent neural networks for machine translation tasks. Instead of distinguishing human generated utterances from machine generated utterances, Lin *et al.* [20] propose to analyze and rank a collection of human generated and machine generated utterances. The generator is trained to produce utterances that achieve higher ranking score than human generated utterances.

III. The Proposed Method

In this work, we propose a seq2seq model with a discriminator network that uses different rewards for the dialogue

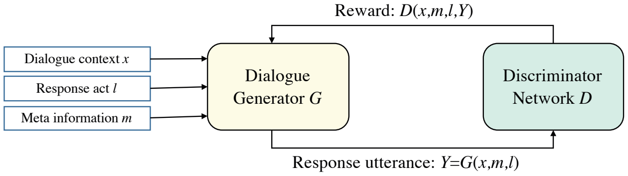

Figure 3. The overall architecture of our proposed conditional dialogue generation framework.

generation tasks. In addition, our proposed model can reduce the training time dramatically, while achieving very competitive performance.

A. *Conditional Dialogue Generation*

In conditional dialogue generation tasks, we are given the dialogue context, including the dialogue history, conversational floor, and meta information. We focus on dialogue utterance in a window size of $k+1$, which contains $k+1$ utterances $[u_i, \ldots, u_{i+k}]$. Then the dialogue history contains k utterances, and the utterance u_{i+k} is the response utterance with N tokens $u_{i+k} = y = [y_1, \ldots, y_N]$. The conversational floor indicates whether the utterances are from the same person. The meta information m contains gender information, education level, etc. Each utterance in the dataset has a dialog act label l such as “statement-opinion”, “agree/accept”, “reject”, etc. Our goal is to train a deep generative model $G_\theta(\cdot)$ to optimize the utterance distribution on the training dataset, where G is parameterized by θ. The model can generate a response utterance $u_{i+k} = G_\theta([u_i, \ldots, u_{i+k-1}], m, l)$ with a specific act label for the given context utterances. Therefore, the proposed model can achieve diverse dialogue generation by providing different dialog acts.

B. *Overall Architecture*

Our overall architecture follows the SeqGAN [7] framework as illustrated in Figure 3. The generator network $G_\theta(\cdot)$ is trained to generate the response utterance with a specific label l automatically. The discriminator network takes dialogue contexts and the response utterance as input and evaluates the quality of the entire response utterance. Since text tokens are discrete, gradients cannot be back-propagated through the discriminator and generator networks. Thus we employ the policy gradient approach to train the generator.

C. *Discriminator*

As illustrated in [21], maximizing the likelihood suffers from the exposure bias problem during inference procedures. We employ the SeqGAN [7] and evaluate the entire response utterance by a discriminator network D [22]. The discriminator is required to distinguish real response utterances and generated response utterances. In conditional dialogue generation tasks, the discriminator network is also required to identify whether the model generates a given dialog act response. Therefore, the objective function of discriminator network consists of two components. The first component is a binary classifier to distinguish real response utterances and generated response utterances. The loss function of the first component is:

$$\mathcal{L}_{global} = \mathbb{E}_y[\log D_g(x, y, m)]. \tag{1}$$

The second component is a multi-class classifier to identify the dialog act labels of response utterances. The loss function of the second component is:

$$\mathcal{L}_{cls} = \mathbb{E}_y[\log D_c(l|x, y, m)] \tag{2}$$

Therefore, the objective function of the discriminator network can be expressed as:

$$\mathcal{L} = \mathcal{L}_{global} + \mathcal{L}_{cls}. \tag{3}$$

D. *Policy Gradient*

If we use the discriminator network to evaluate the quality of the entire response utterance, gradients cannot be back-propagated directly since text tokens are discrete. We employ the policy gradient method to optimize the generator network. Conditional dialogue generation tasks can be considered as a reinforcement learning problem [23], where the state corresponds to the context input and tokens that have been generated; actions correspond to taking a word from vocabulary; and policy is the generator network. The reward in conditional dialogue generation tasks can be provided by the discriminator network. The objective of the policy network is to generate a response utterance that maximizes the expected reward as:

$$J(\theta) = \mathbb{E}_{y \sim G(x,m,l)} R(x, y, m, l), \tag{4}$$

where R is the reward given by a sum probability of the response being a human generated response and correct dialog act. We update the parameter by

$$\theta = \theta + \alpha \nabla J(\theta), \tag{5}$$

where

$$\begin{aligned} \nabla J &\approx R(x, y, m, l) \nabla G_\theta(x, m, l) \\ &= R(x, y, m, l) G_\theta(x, m, l) \log \nabla G_\theta(x, m, l). \end{aligned}$$

E. *Fast Reward Calculation*

In the GAN framework, the discriminator network can only compute the reward for an entire utterance. The reward is shared by all the actions in the generated utterance. However, it is of great importance to assign each action a different reward. When the discriminator assigns the utterance a low reward, this reward is shared by all actions. In this case, part of the actions in the utterance may be appropriate, and they should result in a higher reward. We should assign reward for each action when generating a response utterance.

SeqGAN was proposed to apply Monte Carlo search with a roll-out policy G_θ to evaluate the reward for each action.

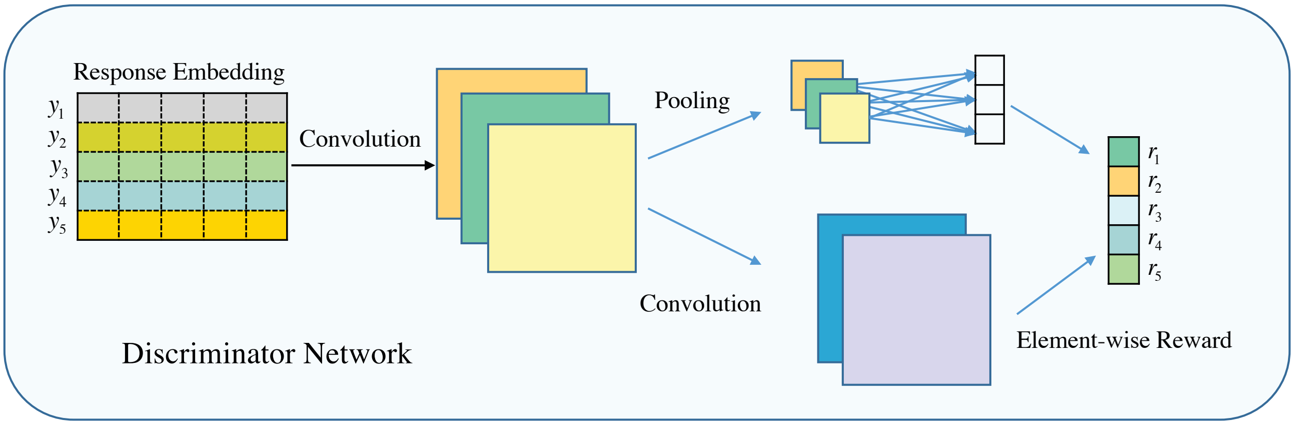

Figure 4. Diagram of the discriminator network. We develop a discriminator network to assign a reward for each token in the response utterance. The loss function of discriminator consists of three components, including a global component, a classification loss and a local loss.

For a given response utterance with N tokens, SeqGAN samples the tokens with the same prefix M times using the policy G_θ when evaluating the reward for the $t-$th token. Therefore, Monte Carlo search must generate the entire utterance $N \times M$ times for an utterance with N tokens. The training procedure of SeqGAN is time-consuming due to the Monte Carlo search.

To calculate the reward for each action efficiently, we propose to train a local discriminator network to evaluate the utterance. The original discriminator employs a classifier to evaluate the real and generated utterances. In order to calculate the reward for each token in the utterance, we can consider each evaluation procedure as a classification task. Therefore, all the tokens in the real response utterances are positive examples, and all the tokens in the generated response utterances are negative examples. We employ convolutional layers with padding in the discriminator network without pooling layers. The spatial size of the discriminator output is the same as that of the input. The output channel of the last convolutional layer is set to two. Then the discriminator network can conduct a binary classification and compute the individual reward for each token in the response utterance. The loss function of local discriminator can be expressed as

$$\mathcal{L}_{local} = \sum_{y_t} \mathbb{E}_{y_t}[\log D(x, y_t, m, l)]. \tag{6}$$

The individual reward of each token can be computed by one forward computation. Compared with the Monte Carlo search method, the training procedure is very efficient by employing the local discriminator network. When computing the reward, we consider both the entire reward and individual reward. The loss function of the discriminator can be expressed as

$$\mathcal{L} = \mathcal{L}_{global} + \mathcal{L}_{cls} + \mathcal{L}_{local}. \tag{7}$$

F. Generator

Our generator network follows an encoder-decoder framework. The input data contains context history $[u_i, \ldots, u_{i+k-1}]$, meta information m and conversational floor. The encoder network extracts features from inputs. The features are then fed into the decoder network as the initial state. We employ recurrent neural networks (RNNs) [24] in both the encoder and decoder networks. Suppose the context history contains T tokens $[u_i, \ldots, u_{i+k-1}]$, and we concatenate all the tokens together as $x = [x_1, \ldots, x_t, \ldots, x_T]$. We use gated recurrent unit (GRU) [25] which can alleviate the vanishing and exploding gradient problems [26] in this work to encode each token as:

$$\begin{aligned} z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z) && (8)\\ r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r) && (9)\\ h_t^{'} &= \tanh(W_h x_t + U_h(r_t \circ h_{t-1}) + b_h) && (10)\\ h_t &= z_t \circ h_{t-1} + (1 - z_t) \circ h_t^{'}, && (11) \end{aligned}$$

where $\circ$ represents the element-wise product operation, W, U, and b are parameters. To obtain a better representation of tokens in the context utterances, we use a bidirectional GRU (BiGRU) [27] as a context encoder to represent x_t as $[\overrightarrow{h_t}, \overleftarrow{h_t}]$ by concatenating the forward hidden state and backward hidden state as:

$$\begin{aligned} \overrightarrow{h_t} &= \overrightarrow{GRU}(x_t, \overrightarrow{h_{t-1}}) && (12)\\ \overleftarrow{h_t} &= \overleftarrow{GRU}(x_t, \overleftarrow{h_{t-1}}), && (13) \end{aligned}$$

where $\overrightarrow{h_t}$ is the hidden state of forward pass in BiGRU, and $\overleftarrow{h_t}$ is the hidden state of backward pass in BiGRU at time t. Then the context representation $[\overrightarrow{h_t}, \overleftarrow{h_t}]$ is fed into another GRU encoder. The GRU network encodes the tokens in k utterances with the corresponding conversation floor and represents the utterances using the last hidden state h_e of the GRU network.

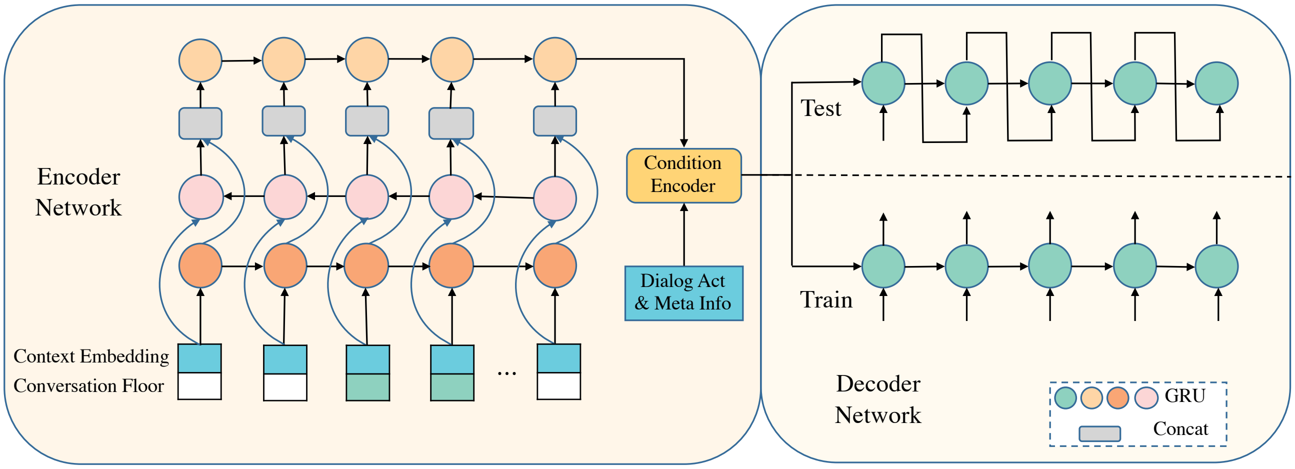

Figure 5. Diagram of our proposed model for conditional dialogue generation tasks. In our model, a bidirectional gated recurrent neural network is used to represent the word token as a context encoder. Then the extracted features are fed in an encoder network using GRU. The hidden state of the last unit is concatenated with meta feature and act label, and the combined feature is used to initialize the decoder hidden state using a fully connected layer. The decoder network employs a GRU to generate the response utterance.

The decoder employs a GRU network to generate a response utterance corresponding to the input context. The decoder network represents the response utterance as the hidden state of a GRU as:

$$h_t = GRU(h_{t-1}, y_t), \tag{14}$$

where g is a gated recurrent unit, h_t is the hidden representation of the $t-$th token in the response utterance, y_t is the $t-$th token, and h_{t-1} is the hidden representation of the previous input. The initial state h_0 of decoder network is $h_0 = W_0[h_e, m, l] + b_0$ by considering all the given input. In addition, a softmax layer is employed to produce the output tokens for the response utterance as:

$$p(y_{t+1}|y_1, \ldots, y_t) = \text{softmax}(b + Wh_t), \tag{15}$$

where b is the bias and W is the projection matrix. The encoder-decoder network can be trained by maximizing the likelihood as:

$$p_G(y|x, m, l) = \Pi_t p_G(x, m, l, y_1 : y_{t-1}). \tag{16}$$

G. Training

We first maximize the likelihood to pre-train the generator network $G_\theta(\cdot)$ using the training dataset. Then the pre-trained network $G_\theta(\cdot)$ is used to generate the response utterance as negative examples to train the discriminator. When the pre-training of generator and discriminator is completed, we employ the discriminator to evaluate the reward for each token in the response utterance and train the generator using policy gradient approach. The generator and discriminator are trained alternatively to achieve better performance. The algorithm is summarized in Algorithm (1).

Algorithm 1: Training procedure for our proposed method.

```
• Pre-train the generator using equation (16) on
  training dataset.
• Generate the response utterances using the
  pre-trained generator as negative examples.
• Pre-train the discriminator using positive and
  negative examples by equation (7).
while not converged do
    • Generate the response utterance using
      generator $G_\theta(\cdot)$.
    • Calculate the reward for each token in
      response utterance using discriminator.
    • Train the generator using policy gradient by
      equation (5).
    • Generate the response utterances using the
      pre-trained generator as negative examples.
    • Train the discriminator using positive and
      negative examples by equation (7).
```

IV. Experiments

In this section, we conduct a conditional dialogue generation experiment on the Switchboard dataset and compare the performance of our proposed method with that of the sequence to sequence (seq2seq) generator and SeqGAN using Monte Carlo search method.

A. Dataset

We apply our proposed method on the Switchboard 1 Release 2 Corpus dataset [28] and Daily Dialog dataset [29]. The Switchboard dataset contains 2400 two-sided telephone

Table I
THE DIALOG ACT DISTRIBUTION IN THE SWITCHBOARD DATASET.

Dialog act	Percent	Dialog act	Percent
statement-non-opinion	37.64	statement-opinion	12.79
acknowledge	21.54	abandon	12.26
yes-no-question	3.15	agree/accept	2.76
appreciation	2.32	others	7.54

Table II
THE DIALOG ACT DISTRIBUTION IN THE DAILY DIALOG DATASET.

Dialog act	Percent	Dialog act	Percent
inform	45.74	question	28.65
directive	16.34	commissive	9.27

conversations. Each conversation in the dataset is assigned with a topic, and the dataset contains 70 topics. The meta information includes education levels and the genders of speakers. The preprocessing procedure is the same as that in [4], including tokenizing using NLTK [30], removing false tokens, and constructing frequent word vocabulary. In the dataset, each utterance is labeled with dialog acts [31], and there are 42 different dialog acts in the Switchboard dataset. The Daily Dialog dataset contains 13,118 multi-turn human-human dialogue annotated with dialog acts. There are 4 different dialog acts in the Daily Dialog dataset. The dialog act distributions for the two datasets have been shown in Tables I and II.

B. Experimental Settings

In the experiment, the context window size is set to 10. All tokens in utterances are processed by a word embedding layer. The size of word embedding is set to 200. We initialize the word embedding with the Glove embedding [32], which is pre-trained on Twitter dataset. The hidden size of context encoder is set to 600. The hidden sizes of encoder and decoder networks are set to 300 and 400, respectively. We employ the Adam [33] optimization method to train the encoder-decoder network. The learning rate of Adam is set to 0.001, and the gradient clip is set to 5. The batch size is set to 30 for training. All the weights in the network are initialized from a uniform distribution in $[-0.08, 0.08]$.

Our discriminator follows the convolutional neural network architecture [22], which contains convolution, max-pooling, and fully connected layers. We employ 1D convolutional layers with kernel sizes of 2, 3, 4, and 5 to extract features at different scales. The number of output channels is set to 200 for all the convolutional layers. The outputs of different convolutional layers are concatenated together and processed by a pooling layer. We apply a dropout layer after the pooling layer, and the dropout rate is set to 0.4. The numbers of output nodes in fully connected layers are set to 2 and number of dialog act (42 for Switchboard dataset and 4 for Daily Dialog dataset) to compute the global loss and classification loss, respectively. We apply a convolutional layer on the concatenated feature map to compute the local loss.

C. Evaluation

Evaluating the quality of generated responses is a challenge for dialogue generation tasks. In this work, we focus on a conditional dialogue generation task in which the generated response utterances must belong to the given dialog act. Therefore, we employ a dialog act classifier to evaluate our generated responses. If our generated responses belong to the given dialog act, the classifier can identify the dialog act correctly. In addition, we also evaluate our response utterances with the BLEU score. The Switchboard dataset only provides one response utterance for a given context input. However, there are multiple reasonable response utterances corresponding to a given context input. In order to evaluate the response utterance with multiple references, we use the information retrieval technology in [4] to construct a reference list, which contains 10 candidates from the same topic. We compute the BLEU1 to 4 [34] to evaluate the similarity between the generated utterances and references.

D. Accuracy of Dialog Act

We quantitatively evaluate the generated response utterances by measuring the accuracy of dialog act. We employ a deep convolutional network [22] for dialog act classification. The model is trained on the training dataset to recognize the dialog act of each utterance, and it achieves 82.55% accuracy on the Switchboard test dataset and 77.35% on the Daily Dialog test dataset for dialog act classification, respectively. We generate response utterances on the test dataset using the seq2seq generator, SeqGAN with Monte Carlo search, and our proposed method.

The results of classification accuracy are shown in Table VI. We can observe from the results that both SeqGAN and our proposed method achieve better classification accuracy than seq2seq generator. The discriminator network in SeqGAN and our proposed model can provide the reward of classification accuracy. The reward can encourage the generator to produce response utterances with a specific dialog act. The classification accuracy of our proposed model is very close to that of the SeqGAN using Monte Carlo search. However, Monte Carlo search is very time-consuming. In the experiment, the rollout number of Monte Carlo search is set to 10. For a response utterance with N tokens, the Monte Carlo search approach takes $10 * N$ times forward propagation computations to assign each token an individual reward. Our proposed method can complete this procedure in one forward propagation, and our computation time is independent of the length of response utterances.

We extract features from the deep convolutional network for dialog act classification and visualize these features using TSNE [35]. The visualization results are shown in Figure 6 and Figure 7 for the two datasets. There are 42 dialog acts

Table III
THE RESPONSE UTTERANCES GENERATED BY DIFFERENT MODELS FOR THREE EXAMPLES ON THE SWITCHBOARD DATASET. ONLY ONE UTTERANCE IS SHOWN IN EACH CASE DUE TO SPACE CONSTRAINTS.

Example 1-Topic: NEWS MEDIA
Context: Well, I can discuss a lot of the news that we have but I feel that that's not part of our responsibility.
Target (acknowledge): Right.

Model	**Response utterance**
seq2seq generator	Yeah.
SeqGAN with MC-search	Yeah.
Our proposed method	Right.

Example 2-Topic: TRIAL BY JURY
Context: I don't know. I look at the people who get excused and it's supposed to be a jury of your peers and I'm not sure it always is.
Target (statement-opinion): And it is a huge burden to put on people who don't know the law.

Model	**Response utterance**
seq2seq generator	And it's not that it's not going to be a deterrent.
SeqGAN with MC-search	It's not that bad that it's not fair to have to have to go ahead and get it.
Our proposed method	And i don't know if it's a problem or not but I think it's a law that's important.

Example 3-Topic: AIR POLLUTION
Context: And you know they say that the coal we burn here in Indiana is responsible for a large part of the acid rain that they have up in New York state and in Canada and all that.
Target (statement-opinion): And it's just you know it's ridiculous when we could be spending the money to develop the technology to you know clean up some of these.

Model	**Response utterance**
seq2seq generator	You know it's not that bad that they're not doing that they're not going to be able to handle it.
SeqGAN with MC-search	And I think that's what they're doing it's not going to be a problem that they're going to be aware of it.
Our proposed method	And I think the air conditioning is part of the issue of anyone.

Table IV
THE RESPONSE UTTERANCES GENERATED BY DIFFERENT MODELS FOR TWO EXAMPLES ON THE DAILY DIALOG DATASET. ONLY ONE UTTERANCE IS SHOWN IN EACH CASE DUE TO SPACE CONSTRAINTS.

Context: **A**: How may I help you? **B**: Could you cancel a check for me? **A**: What's the problem ? **B**: I wrote the check out for way too much.
Target (question): How much ?

Model	**Response utterance**
seq2seq generator	Ok. Here you go.
SeqGAN with MC-search	How much do you owe me ?
Our proposed method	How much do you have for ?

Context: **A**: Really ? I thought Barcelona was going to win. They are a much better team than Madrid. **B**: If Barcelona had played as they normally do, they should have won. what happened? **A**:Their goalkeepers got injured after the first goal, but he kept on playing anyway. **B**: That explains it .
Target (question): How many more games are left this season?

Model	**Response utterance**
seq2seq generator	How many more games are we playing ?
SeqGAN with MC-search	How many more games are left this season ?
Our proposed method	How many more games are left ?

Table V
THE RESPONSE UTTERANCES GENERATED FOR A GIVEN CONTEXT HISTORY USING DIFFERENT DIALOG ACTS. ONLY ONE UTTERANCE IS SHOWN IN EACH CASE DUE TO SPACE CONSTRAINTS.

Context: **A**: Why not go again to celebrate out one - year anniversary ? We can go to the same beach , stay in the same hotel and enjoy a dinner in the same restaurant . **B**: Are you kidding ? Can you afford it ? Do you think we can get a room on such short notice ? **A**: Never mind that , I'll take care of it. Are you available next week ? **B**: Yeah , I think so . **A**: Ok . I'll make the arrangements. It will be great .
Target (inform): Wonderful ! I'll start packing our suitcases.

Act	**Response utterance**
inform	Great ! I'll take it .
question	Great ! That'll do my best . How do we start doing this house ?
commissive	Great ! I'll start looking forward to help about hotels .

in the Switchboard dataset. We only visualize the top 4 frequent dialog acts. We can observe from the results that the response utterances with the same dialog act are close to each other. The visualization results demonstrate that our model can generate response utterances with a desired dialog act.

Table VI
CLASSIFICATION ACCURACY OF EACH MODEL ON THE SWITCHBOARD AND DAILY DIALOG TEST DATASET. THE HIGHEST CLASSIFICATION ACCURACY IS IN BOLD.

DataSet	Baseline	MC-search	Our model
Switchboard	77.74	**83.99**	83.16
Dailydialog	76.21	76.35	**77.76**

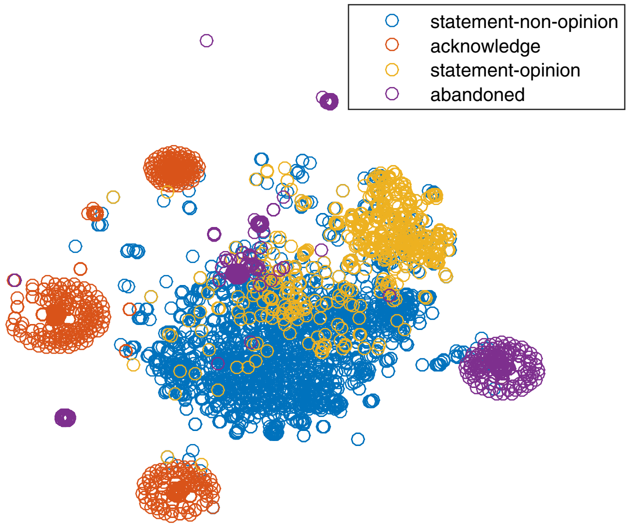

Figure 6. TSNE visualization of dialog act for test response utterances with top 4 frequent dialog acts in the Switchboard dataset. The features are extracted from the deep convolutional network for dialog act classification.

E. Time Complexity Analysis

We compare the training time of Monte Carlo search and that of our proposed method. These models are trained by using NVIDIA GeForce GTX Titan XP Graphics Cards, and the training time is shown in Table VII. The rollout number for Monte Carlo search is set to 10 in the experiments. If the length of all utterances in the dataset is equal to 10, Monte Carlo search costs 100 times more training time than our proposed method.

F. Similarity Analysis

When the response utterances contain some key tokens, the key tokens can encourage the classifier to produce correct predictions. Therefore, we compute the BLEU score to evaluate the similarity between response utterances and references. Table VIII and Table IX shows the results of BLEU score on Switchboard and Daily Dialog dataset. We can observe from the results that our proposed model can achieve better performance than seq2seq generator.

G. Qualitative Analysis

We randomly select generated response utterances on the Switchboard and Daily Dialog test dataset using seq2seq generator, SeqGAN with Monte Carlo search and our proposed model. The response utterances are shown in Table III for Switchboard dataset and Table IV for Daily Dialog dataset. From Table III, We can observe that the three models can generate good response utterances when the dialog act is easy to represent such as acknowledge, accept. If the model is required to generate more complex response utterances, SeqGAN and our proposed model can achieve better performance. In the third example in Table III, the seq2seq model generates good tokens at the beginning of the utterance. When the seq2seq model generates an undesired token, it suffers from the “exposure bias” problem and fails to generate good utterances afterwards. SeqGAN and our proposed model are trained with discriminators to alleviate the effect of “exposure bias” problem. Therefore, these two models can outperform the seq2seq model. Our proposed model is more time-efficient than the Monte Carlo search approach.

We can observe from Table IV that our proposed model can generate response utterances corresponding to the given dialog acts when employing the classification reward in the discriminator network. For the first example in Table IV, the given dialog act is “question”. The seq2seq model generates a response utterance which belongs to “inform”. Since we employ classification reward in SeqGAN and our proposed model, these two models can provide response utterances with correct dialog act. Compared with MC-search method, our proposed method can generate response utterances with a better understanding of context history. For the second exam-

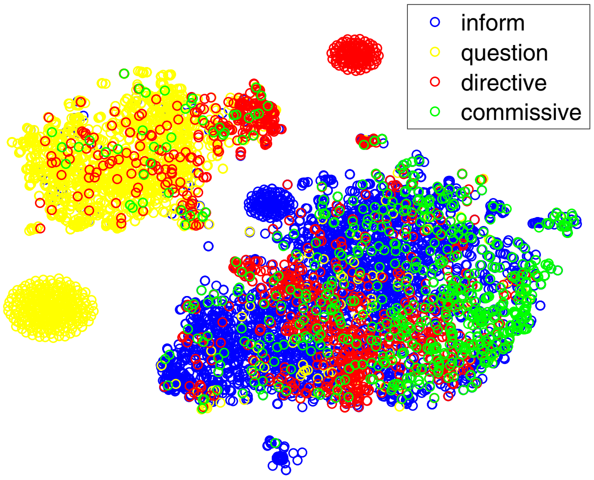

Figure 7. TSNE visualization of dialog act for test response utterances in the Daily Dialog dataset. The features are extracted from the deep convolutional network for dialog act classification.

Table VII
COMPARISON OF TRAINING TIME BETWEEN THE MONTE CARLO SEARCH AND OUR PROPOSED METHOD FOR ONE EPOCH.

Metrics	MC-search	Our model
Time	62.5 hour	**0.4** hour

Table VIII
PERFORMANCE OF EACH MODEL ON SIMILARITY MEASURES FOR SWITCHBOARD DATASET. THE HIGHEST SCORE IN EACH ROW IS IN BOLD. NOTE THAT OUR BLEU SCORES ARE NORMALIZED TO $[0, 1]$.

Metrics	Baseline	Our model	MC-search
BLEU-1	**0.4561**	0.4558	0.4527
BLEU-2	0.3937	**0.3971**	0.3946
BLEU-3	0.3506	**0.3560**	0.3495
BLEU-4	0.3384	**0.3453**	0.3336

Table IX
PERFORMANCE OF EACH MODEL ON SIMILARITY MEASURES FOR DAILY DIALOG DATASET. THE HIGHEST SCORE IN EACH ROW IS IN BOLD. NOTE THAT OUR BLEU SCORES ARE NORMALIZED TO $[0, 1]$.

Metrics	Baseline	Our model	MC-search
BLEU-1	0.4830	**0.4900**	0.4875
BLEU-2	0.3644	**0.3725**	0.3693
BLEU-3	0.2736	**0.2829**	0.2784
BLEU-4	0.2090	**0.2193**	0.2135

ple in Table IV, all the three models can generate response utterances that belong to the "question" category. However, the seq2seq model generates the response utterance without understanding the context history. We employ the discriminator to distinguish human generated and machine generated response utterances. The discriminator network encourages the model to generate response utterances that are close to the human generated utterances. Therefore, our proposed model can generate better response utterances based on the given context history.

Providing diverse response utterances is of great importance for dialogue generation models. Our proposed model can generate different response utterances based on the given dialog acts. The results have been shown in Table V. We can observe from the results that our proposed model can provide different response utterances based on the given dialog acts.

V. CONCLUSION

In this work, we propose to employ an encoder-decoder model to tackle conditional dialogue generation tasks. The model takes the context history, meta information, and dialog act as input and generates a response utterance with a desirable dialog act. To alleviate the effect of "exposure bias" problem, we employ a SeqGAN framework and evaluate the entire utterance by considering the global similarity and the dialog act. The SeqGAN uses Monte Carlo search to assign each token in the utterance an individual reward, which is very time-consuming for the training procedure. We propose a local discriminator network to compute the individual reward for each token by one forward propagation. Experimental results demonstrate that our proposed model can achieve similar performance, while reducing training time dramatically.

ACKNOWLEDGMENT

This work was supported in part by National Science Foundation grants IIS-1908198 and DBI-1661289.

REFERENCES

[1] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," in *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2016, pp. 110–119.

[2] J. Li, W. Monroe, A. Ritter, D. Jurafsky, M. Galley, and J. Gao, "Deep reinforcement learning for dialogue generation," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1192–1202.

[3] J. Li, W. Monroe, T. Shi, S. Jean, A. Ritter, and D. Jurafsky, "Adversarial learning for neural dialogue generation," *arXiv preprint arXiv:1701.06547*, 2017.

[4] T. Zhao, R. Zhao, and M. Eskenazi, "Learning discourse-level diversity for neural dialog models using conditional variational autoencoders," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2017, pp. 654–664.

[5] T. Zhao, K. Lee, and M. Eskenazi, "Unsupervised discrete sentence representation learning for interpretable neural dialog generation," *arXiv preprint arXiv:1804.08069*, 2018.

[6] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *Advances in neural information processing systems*, 2014, pp. 3104–3112.

[7] L. Yu, W. Zhang, J. Wang, and Y. Yu, "Seqgan: Sequence generative adversarial nets with policy gradient." in *AAAI*, 2017, pp. 2852–2858.

[8] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction.* MIT press Cambridge, 1998, vol. 1, no. 1.

[9] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in neural information processing systems*, 2000, pp. 1057–1063.

[10] M.-T. Luong and C. D. Manning, "Achieving open vocabulary neural machine translation with hybrid word-character models," in *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, vol. 1, 2016, pp. 1054–1063.

[11] S. Wiseman and A. M. Rush, "Sequence-to-sequence learning as beam-search optimization," in *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, 2016, pp. 1296–1306.

[12] R. Nallapati, B. Zhou, C. dos Santos, C. Gulcehre, and B. Xiang, "Abstractive text summarization using sequence-to-sequence rnns and beyond," in *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, 2016, pp. 280–290.

[13] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Acoustics, speech and signal processing (icassp), 2013 ieee international conference on*. IEEE, 2013, pp. 6645–6649.

[14] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *Advances in neural information processing systems*, 2014, pp. 2672–2680.

[15] E. L. Denton, S. Chintala, R. Fergus *et al.*, "Deep generative image models using a laplacian pyramid of adversarial networks," in *Advances in neural information processing systems*, 2015, pp. 1486–1494.

[16] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1125–1134.

[17] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," *arXiv preprint*, 2016.

[18] L. Cai, Z. Wang, H. Gao, D. Shen, and S. Ji, "Deep adversarial learning for multi-modality missing data completion," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1158–1166.

[19] Z. Yang, W. Chen, F. Wang, and B. Xu, "Improving neural machine translation with conditional sequence generative adversarial nets," *arXiv preprint arXiv:1703.04887*, 2017.

[20] K. Lin, D. Li, X. He, Z. Zhang, and M.-T. Sun, "Adversarial ranking for language generation," in *Advances in Neural Information Processing Systems*, 2017, pp. 3155–3165.

[21] S. Bengio, O. Vinyals, N. Jaitly, and N. Shazeer, "Scheduled sampling for sequence prediction with recurrent neural networks," in *Advances in Neural Information Processing Systems*, 2015, pp. 1171–1179.

[22] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[23] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," in *Reinforcement Learning*. Springer, 1992, pp. 5–32.

[24] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[25] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[26] I. Goodfellow, Y. Bengio, A. Courville, and Y. Bengio, *Deep learning*. MIT press Cambridge, 2016, vol. 1.

[27] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[28] J. Godfrey and E. Holliman, "Switchboard-1 release 2: Linguistic data consortium," *SWITCHBOARD: A User's Manual*, 1997.

[29] Y. Li, H. Su, X. Shen, W. Li, Z. Cao, and S. Niu, "Dailydialog: A manually labelled multi-turn dialogue dataset," in *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, vol. 1, 2017, pp. 986–995.

[30] S. Bird, E. Klein, and E. Loper, *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.", 2009.

[31] A. Stolcke, K. Ries, N. Coccaro, E. Shriberg, R. Bates, D. Jurafsky, P. Taylor, R. Martin, C. V. Ess-Dykema, and M. Meteer, "Dialogue act modeling for automatic tagging and recognition of conversational speech," *Computational linguistics*, vol. 26, no. 3, pp. 339–373, 2000.

[32] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, 2014, pp. 1532–1543.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[34] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu, "Bleu: a method for automatic evaluation of machine translation," in *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 2002, pp. 311–318.

[35] L. v. d. Maaten and G. Hinton, "Visualizing data using t-sne," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.