Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)

Reinforcing Coherence for Sequence to Sequence Model in Dialogue Generation

Hainan Zhang, Yanyan Lan, Jiafeng Guo, Jun Xu and Xueqi Cheng
University of Chinese Academy of Sciences, Beijing, China
CAS Key Lab of Network Data Science and Technology,
Institute of Computing Technology, Chinese Academy of Sciences
zhanghainan@software.ict.ac.cn, {lanyanyan, guojiafeng, junxu, cxq}@ict.ac.cn

Abstract

Sequence to sequence (Seq2Seq) approach has gained great attention in the field of single-turn dialogue generation. However, one serious problem is that most existing Seq2Seq based models tend to generate common responses lacking specific meanings. Our analysis show that the underlying reason is that Seq2Seq is equivalent to optimizing Kullback–Leibler (KL) divergence, thus does not penalize the case whose generated probability is high while the true probability is low. However, the true probability is unknown, which poses challenges for tackling this problem. Inspired by the fact that the coherence (i.e. similarity) between post and response is consistent with human evaluation, we hypothesize that the true probability of a response is proportional to the coherence degree. The coherence scores are then used as the reward function in a reinforcement learning framework to penalize the case whose generated probability is high while the true probability is low. Three different types of coherence models, including an unlearned similarity function, a pretrained semantic matching function, and an end-to-end dual learning architecture, are proposed in this paper. Experimental results on both Chinese Weibo dataset and English Subtitle dataset show that the proposed models produce more specific and meaningful responses, yielding better performances against Seq2Seq models in terms of both metric-based and human evaluations.

1 Introduction

This paper focuses on the problem of single-turn dialogue generation, which is expected to automatically generate an appropriate response for a given post. Following conventional data-driven generation framework of statistical machine translation, most existing neural conversation models are based on a Seq2Seq architecture [Sutskever *et al.*, 2014]. In these models, a recurrent neural network (RNN) encoder is first utilized to encode the input post to a vector, and another RNN decoder is then used to generate the response. To learn the model parameters, a maximum likelihood estimation approach is applied on the training data which consists of many post-response pairs. The intrinsic philosophy is that the true probability would be estimated by the generated probability with proper parameters.

Though Seq2Seq has the ability to generate fluent responses, one serious problem is that the generated responses are usually common, such as '*I do not know*', '*What does this mean?*' and '*Haha*' [Li *et al.*, 2016a; Mou *et al.*, 2017]. Clearly, these kinds of responses lack specific meanings for further widening and deepening of the dialogue, which will have a bad effect on the users' experience. Through our analysis, the main reason is that the objective of Seq2Seq is equivalent to minimizing the KL divergence between the generated probability and the true probability. However, KL divergence is not symmetric, thus it will not penalize the case whose generated probability is high while the true probability is low, which is exactly the case of common responses.

In this paper, we propose to utilize the coherence (i.e. similarity) between the generated responses and the original post as an estimation of the true probability, with inspiration comes from the fact that the similarity measure between post and response embeddings is consistent with human evaluation. Specifically, three kinds of coherence models are adopted in this paper. Firstly, an unlearned similarity function, such as cosine similarity, can be directly used as the coherence model. Secondly, the previous semantic text matching models can be regarded as good candidates for measuring the coherence between a post and its corresponding response. In this paper, we use two pretrained matching functions, i.e., GRU bilinear model [Socher *et al.*, 2013] and MatchPyramid [Pang *et al.*, 2016], which are representatives of two different kinds of deep matching models, i.e., representation focused methods and interaction focused methods. Thirdly, an end-to-end dual learning architecture similar to [Xia *et al.*, 2016] can be adopted to jointly learn the parameters of response generation model and coherence model. After that, the coherence model is used as the reward function in a reinforcement learning framework for optimization, which will guide the learning process to penalize the case whose generated probability is high while the true probability is low.

We evaluate the proposed models on two public datasets, i.e. the Chinese Weibo and the English Subtitle dataset. Experimental results show that our models significantly outper-

form traditional Seq2Seq models and their variations with respect to both metric-based evaluations [Li *et al.*, 2016a; Shen *et al.*, 2017] and human judgments.

2 Related Work

The basic neural-based Seq2Seq framework for dialogue generation is inspired by the studies of statistical machine translation (SMT). Sutskever et al. [Sutskever *et al.*, 2014] proposed the original Seq2Seq framework, which used a multi-layered Long Short-Term Memory(LSTM) to map the input sequence to a fixed dimension vector and then used another LSTM to decode the target sequence from the vector. At the same time, Cho et al. [Cho *et al.*, 2014] followed the above architecture, and proposed to feed the last hidden state of encoder to every cell of decoder, which enhanced the influence of contexts in generation. To further alleviate the long dependency problem, Bahdanau et al. [Bahdanau *et al.*, 2015] introduced the attention mechanism into the neural network and achieved encouraging performances. Many studies [Shang *et al.*, 2015; Vinyals and Le, 2015] applied the above neural SMT models to dialogue generation, and gained promising performances.

Although the current Seq2Seq model is capable to generate fluent responses, these responses are usually general, such as *'I don't know'*,*'interesting'*, and *' what is it'*. Li et al. [Li *et al.*, 2016a] proposed a mutual information model to tackle this problem. However, it is not a unified training model, instead it still trained maximum likelihood model, and used the Maximum Mutual Information criterion only for testing to rerank the primary top-n list generated by Seq2Seq. Mou et al. [Mou *et al.*, 2017] proposed a forward-backward keyword method which used a pointwise mutual information to predict a noun as a keyword and then used two Seq2Seq models to generate the forward sentence and the backward sentence. Xing et al. [Xing *et al.*, 2017] proposed a joint attention mechanism model which modified the generation probability by adding the topic keywords likelihood to the generated maximum likelihood. It has to train an extra LDA model from an extra corpus to generate the topic keyword candidates. However, if the new posts are not in these topics, the user has to re-trained the LDA model to adapt the new data. Recently, SeqGAN [Yu *et al.*, 2017] and Adver-REGS [Li *et al.*, 2017] tried to use GAN for generation, where the discriminator score is used as a reward function. However, the meaning of this kind of reward function is not clear.

3 Sequence to Sequence Framework

We first introduce the typical LSTM-based Seq2Seq model [Bahdanau *et al.*, 2015] used in dialogue generation.

Given a post $X=\{x_1, \ldots, x_M\}$ as the input, a standard LSTM first maps the input sequence to a fixed-dimension vector h_M as follows.

$$\begin{aligned} i_k &= \sigma(W_i[h_{k-1}, w_k]), \;\; f_k = \sigma(W_f[h_{k-1}, w_k]), \\ o_k &= \sigma(W_o[h_{k-1}, w_k]), \;\; l_k = \tanh(W_l[h_{k-1}, w_k]), \\ c_k &= f_k c_{k-1} + i_k l_k, \qquad h_i = o_k \tanh(c_k), \end{aligned}$$

where i_k, f_k and o_k is the input, memory, and output gate, respectively. w_k is the word embedding for x_k, and h_k stands for the vector computed by LSTM at time k by combining w_k and h_{k-1}. c_k is the cell at time k, and σ denotes the sigmoid function. W_i, W_f, W_o and W_l are parameters.

Then another LSTM is used as the decoder to map the vector h_M to the ground-truth response $Y=\{y_1, \cdots, y_N\}$. Given the context vector h_M and the previous generated words $\{y_1, \ldots, y_{i-1}\}$, the decoder is typically trained to predict the next word y_i. In other words, the decoder defines a probability over the output Y by decomposing the joint probability into conditionals by the chain rule in probability theory.

Usually the attention mechanism is further introduced to the above Seq2Seq framework. Instead of directly using h_M as the context vector in the decoder, we let the new context vector, denoted as s_i, to be dependent on the whole sequence $(h_1, \cdots, h_M)$, where each h_k contains information about the input sequence with a strong focus on the parts surrounding the k-th word of the input sentence. Specifically, the context vector s_i is usually defined as a weighted sum of these h_k:

$$s_i = \sum_{k=1}^{M} \alpha_{ik} h_k.$$

The weight α_{ik} of each representation h_k is computed as:

$$\begin{aligned} \alpha_{ik} &= \frac{\exp(e_{ik})}{\sum_{j=1}^{M} \exp(e_{ij})}, \\ e_{ik} &= v^T \tanh(W_1 h'_{i-1} + W_2 h_k), \end{aligned}$$

where v^T,W_1 and W_2 are learned parameters. e_{ik} is an alignment model which scores how well the inputs around position k and the output at position i match, which is based on the LSTM hidden state h'_{i-1} (just before emitting y_i), and h_k of the input sentence.

Given a set of training data $\mathcal{D} = \{(X, Y)\}$, Seq2Seq assumes that these data are i.i.d. sampled from the probability P_g, and the following negative log likelihood is used as the objective for minimization.

$$\mathcal{L} = -\sum_{(X,Y)\in\mathcal{D}} \log P_g(Y|X). \tag{1}$$

4 Motivation

Though the above Seq2Seq model has the ability to generate fluent responses for a given post, the responses are usually common [Li *et al.*, 2016a; 2016b; 2017]. Through our data analysis, we find that the empirical probabilities of these generated responses are very low. Specifically, we define two metrics, i.e. hit rate and hit probability as follows.

$$HitR = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n}\sum_{j=1}^{n} I((X_i, Y_{ij}) \in \mathcal{D}), \tag{2}$$

$$HitP = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n}\sum_{j=1}^{n} I((X_i, Y_{ij}) \in \mathcal{D}) \times P_e(Y_{ij}|X_i) \tag{3}$$

where $X_i, i = 1, \cdots, m$ stands for the i-th posts in the training data, and $Y_{ij}, j = 1, \cdots, n$ stands for the j-th generated response with respect the post X_i, $I((X_i, Y_{ij}) \in \mathcal{D})$ is an

human score	1	2	3	4	5
cosine similarity	60.19	57.39	60.30	62.06	66.41

Table 1: The average similarity(%) of post and its generated responses with human score on STC.

indictor function, and $P_e(Y_{ij}|X_i)$ is the empirical probability of Y_{ij} given X_i on the training data. That is to say, if there are five ground-truth responses for post X_i in the training data, the empirical probability of each ground-truth response is 20%, and the empirical probabilities of other responses are 0. From the definition, we can see that hit rate reflects the percentage of ground-truth responses generated, while the hit probability considers not only the number but also the probabilities of these ground-truth responses on the training data.

According to our statistics on a benchmark dialogue data STC[1], the hit rate and hit probability for Seq2Seq with attention are 0.004239 and 0.00091 respectively. Therefore, we conclude that most generated responses are not ground-truth responses, and the true probabilities of most generated responses are probably very low. Since the indicator function is very strict and only considers exact matched responses, we can also modify it to include the semantic matched ones. That is to say, for a given generated response Y_{ij}, if there exists a post-response pair (X_i, Y_s) such that the cosine similarity between Y_{ij} and Y_s is sufficiently large (i.e. 0.9 in this paper), $I((X_i, Y_{ij}) \in \mathcal{D}) = 1$. Otherwise $I((X_i, Y_{ij}) \in \mathcal{D}) = 0$. In this case, hit rate and hit probability becomes 0.1449 and 0.01255, respectively. Therefore, even we consider the semantic relations, the true probabilities of generated responses are still very low.

The main reason of the above observation is that Seq2Seq is equivalent to minimizing the following KL divergence between the generated distribution $P_g(Y|X)$ and the true distribution $P_r(Y|X)$, i.e. $KL(P_r(Y|X)||P_gY|X))$, as shown in [Arjovsky *et al.*, 2017]. Since KL divergence is asymmetric, it only penalizes the case when P_g is low and P_r is high, but fails to penalize the case when P_g is high and P_r is low. Specifically, if $P_r(Y|X) > 0$ but $P_g(Y|X) \to 0$, the integrand inside the KL divergence grows quickly to infinity, meaning that this cost functions assigns an extremely high cost to the generated probability if it does not cover parts of the data. However, if $P_g(Y|X) > 0$ but $P_r(Y|X) \to 0$, the value inside the KL goes to 0, meaning that this cost function will pay extremely low cost for generating fake responses. This is accordant with our observation that Seq2Seq tends to generate common responses with high generated probability but low true probability. In order to tackle this problem, we need to consider not only the likelihood of the generated probability $P_g(Y|X)$, but also the true data distribution $P_r(Y|X)$. However, the true probability $P_r(Y|X)$ is usually unknown, which poses great challenges for tackling this problem.

Luckily, we find some insights from the data analysis. Specifically, we make a relevance statistics on STC data of the human evaluation and average similarity between one post and its generated response. For human evaluation, given 300 randomly sampled post and their generated responses, three annotators (all of them are computer science majored students) are required to give 5-graded judgements. The criteria are defined as follows:

1. the response is nonfluent or logically wrong;
2. the response is fluent but not related with the post, including the case when some un-related common response;
3. the response is fluent and weakly related, but it's common which can response many other posts;
4. the response is fluent and strongly related with its post;
5. the response is fluent and strongly related, which is like following a real person's tone.

As shown in Table 1, the similarity measure is consistent with the human evaluation.[2] This finding inspires us the true probability of a response is highly likely to be proportional to the coherence score (i.e. similarities) between this response and its post. Therefore, it is natural to utilize the coherence score to act as a reward function to penalize the case whose generated probability is high while the true probability is low.

5 Coherence Model

In this paper, we propose three kinds of coherent model to compute the similarity between a post and a response: an unlearned similarity function, two pretrained semantic matching functions, and an end-to-end dual learning architecture.

5.1 Unlearned Similarity Function

The simplest way to measure the coherence is to directly use an unlearned similarity function. For example, we can use the cosine function to act as the coherence model.

$$r_{cos}(X, G) = \frac{< h(X), h(G) >}{\|h(X)\|\|h(G)\|},$$

where $< \cdot, \cdot >$ and $\| \cdot \|$ denote the inner product and L_2 norm, respectively. $h(X)$ and $h(G)$ stand for the sentence representations of post and generated response. Following previous practice [Shen *et al.*, 2017], we directly use the mean over the word embeddings in the sentence as the sentence representations in this paper.

5.2 Pretrained Semantic Matching Functions

Recently, extensive semantic text matching models [Pang *et al.*, 2016; Wan *et al.*, 2016b] have been proposed to capture the coherence relationship between two texts. In this paper, we utilize two pretrained matching function, i.e., bilinear model [Socher *et al.*, 2013] with GRU and MatchPyramid [Pang *et al.*, 2016] to measure the coherence. The reason to choose these models is that they are representatives of the two different categories of text matching models [Guo *et al.*, 2016], i.e., representation focused methods and interaction focused methods.

[1] http://ntcirstc.noahlab.com.hk/STC2/stc-cn.htm

[2] In order to eliminate the influence of same X and Y which similarity score is perfect but the human score is not likely to be high, we didn't calculate their similarity score in this results.

GRU Bilinear Model

GRU bilinear model is a typical representation focused semantic matching model, which focuses on learning the semantic representations of each sentence. Given the post X and the generated response G, a GRU model is utilized to encode them to sentence embeddings $h(X)$ and $h(G)$. Then a bilinear function is used to model the transformation from post to response:

$$r_{bi}(X, G) = h(X) \cdot W \cdot h(G),$$

where W is the transformation matrix.

MatchPyramid

MatchPyramid [Pang *et al.*, 2016] is a typical interaction focused semantic matching model, which is capable to capture different levels of interaction signals. For the dialogue generation task, the coherence pattern between post and response may not lie in the global sentence-level, as assumed in cosine similarity and GRU bilinear, but in some local evidences. For example, post and responses usually have some similar keywords or phrases [Mou *et al.*, 2017]. Therefore, MatchPyramid can also be viewed as a suitable coherence model, denoted as r_{mp}. Specifically, the word embedding matrix between X and G is first constructed, and a convolutional neural network is then applied. Finally, a multiple layer perceptron is conducted to output the matching score $r_{mp}(X, G)$.

For pretrainning the above GRU-bilinear and MatchPyramid models, we follow the practice of traditional semantic matching [Wan *et al.*, 2016a] and use a pairwise loss function for optimization. Specifically, for each post and positive (ground-truth) response pair in the training data, we randomly sample five negative responses, chosen from other post's ground-truth responses, to construct the training data. Then the training criterion is to learn appropriate parameters to rank the positive response higher than the negative ones.

5.3 Dual Learning Architecture

Both GRU bilinear model and MatchPyramid are capable to capture the complex coherence relationship between post and response, however, the learning of coherence and maximum likelihood are isolated. In this paper, we propose an end-to-end dual learning architecture to jointly learn the parameters of coherence and maximum likelihood. Dual learning is first proposed by Xia et al. [Xia *et al.*, 2016] for statistical machine translation. We apply it here because it is suitable to fulfill the requirement to represent the coherence between post and generated response, which can be described as two-agent communication process:

1. The first agent understands the post, and sends a message to the second agent, which converts the message from post to response.
2. The second agent understands the response, and measures the coherence of the message from the post and the response, which is propagated to the first agent.
3. According to the coherence measure suggested by the second agent's, the first agent modifies the dialogue generation model.
4. The above three steps can also be started with the second agent to obtain a dual joint learning architecture.

The dual-learning agents are two standard sequence to sequence models: the first agent generates the response for the given post and the second agent generates post for the given response. In the first step, given the post X, the first agent generates the response G_1 and sends G_1 to the second agent. The second agent calculates the conditional probability of $P(X|G_1)$ as its coherence measure and tells the first agent. Finally, the first agent uses the coherence measure to guide the response generation process. The second agent then starts the communication. For the given ground-truth response Y, it first generates the response G_2 and sends it to the first agent. The first agent calculates the conditional probability of $P(Y|G_2)$ as its coherence measure, and tells the second agent. Finally, the second agent uses the coherence measure to guide the post generation process. The above process repeated coordinately until convergence. Therefore, there are two coherence models $r_{dual1}(X, G)$ and $r_{dual2}(X, G)$, which can be defined as follows.

$$\begin{aligned}
&G_1 = \arg\max P_1(G_1|X), \\
&r_{dual1}(X, G_1) = \log P_2(X|G_1), \\
&G_2 = \arg\max P_2(G_2|Y), \\
&r_{dual2}(Y, G_2) = \log P_1(Y|G_2),
\end{aligned}$$

where $P_2(X|G_1)$ is the probability of generating post X given the sentence G_1 generated by the first agent, and $P_1(Y|G_2)$ is the probability of generating response Y given the sentence G_2 by the second agent, where P_1 and P_2 are two different Seq2Seq models.

5.4 Optimization

Inspired by recent work of [Li *et al.*, 2016b; 2017; Yu *et al.*, 2017], we treat the problem of generating high quality responses with maximum likelihood as a reinforcement learning problem, in which the coherence model acts as the reward function and is observed when the model arrives at the end of the sequence. The reinforcement learning model P_{RL} is initialized by the pretrained Seq2Seq with attention model $P_{Seq2Seq}$. Specifically, an action is the dialogue response to generate. The action space is infinite since arbitrary-length sequences can be generated. A state is denoted by the post in the single-turn dialogue generation, which is usually transformed to a vector representation. A policy takes the form of an LSTM encoder-decoder, and is defined by its parameters. Note that we are using a stochastic representation of the policy, i.e., a probability distribution over actions given states. The reward function is defined by the coherence model for each action. For the unlearned similarity function and pretrained semantic matching functions, the reinforcement learning architecture is shown in Figure 1(a). Given an input post X, Seq2Seq first takes an action to generate a response G, and then it obtains the reward $r(X, G)$ from the coherence models. We use the policy gradient methods [Sutton *et al.*, 2000] for optimization, with the expected reward defined as:

$$J(\theta) = \mathbb{E}[r(X, G)]. \tag{4}$$

The gradient is estimated as:

$$\nabla J(\theta) = r(X, G) \nabla \log P_{RL}(G|X), \tag{5}$$

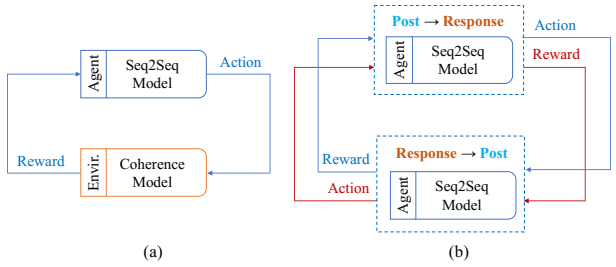

Figure 1: The architecture of reinforcement learning. (a) shows the architecture for unlearned similarity function and pretrained matching functions. (b) shows the architecture for dual learning approach.

Please note that the coherence scores of the pretrained matching functions need to be normalized, due to the varied ranges. In this paper we use min-max normalization for r_{bi} and r_{mp}. Take bilinear model as an example, we randomly select some negative sentences $\{G'_1, \ldots, G'_m\}$ generated from Seq2Seq, and then calculate the scores: $\mathcal{S} = \{r_{bi}(X, G), r_{bi}(X, G'_1), \ldots, r_{bi}(X, G'_m)\}$. The min-max normalized coherence score $r(X, G)$ is then defined as follows.

$$r(X, G) = \frac{r_{bi}(X, G) - \min(p)}{\max(p) - \min(p)},$$

where $\min(p)$ and $\max(p)$ are the minimum value and maximum value of $\mathcal{S}$.

The reinforcement learning process of dual learning architecture is more complex, as shown in Figure 1(b). There are two opposite reinforcement learning processes. For the first reinforcement learning process, the first agent (post→response) generates a response G_1 for a given the input post X, and then obtains the reward r_{dual1} from the second agent. The expected reward and gradient are defined as:

$$J_1(\theta) = \mathbb{E}[r_{dual1}(X, G_1)],$$
$$\nabla J_1(\theta) = r_{dual1}(X, G_1) \nabla \log P_{RL}(G_1|X).$$

For the second reinforcement learning process, the second agent (response→post) generates a post G_2 for a given response Y, and then receives the reward r_{dual2} from the first agent. The expected reward and gradient are defined as:

$$J_2(\theta) = \mathbb{E}[r_{dual2}(Y, G2)],$$
$$\nabla J_2(\theta) = r_{dual2}(Y, G2) \nabla \log P_{RL}(G_2|Y).$$

Both aforementioned gradients are plugged into the policy gradient methods [Sutton *et al.*, 2000] to optimize the reinforcement learning process, shown in Fig. 1.

6 Experiments

6.1 Experimental Settings

Datasets. We use two public datasets in our experiments. The Chinese Weibo dataset, named STC [Shang *et al.*, 2015], consists of 3,788,571 conversational post-response pairs extracted from the Chinese Weibo website and cleaned by the data publishers. We randomly split the data to training, validation, and testing sets, which contains 3,000,000, 388,571 and 400,000 pairs, respectively. We also use an English conversation data, named OpenSubtitles[3] (OSDb) dataset , to test our model. OSDb is a large, open-domain dataset containing roughly 60M-70M scripted lines spoken by movie characters. We randomly selected a subset of OSDb dataset in our experiment, which contains 3,800,000 post-response pairs. We split the data to 3,000,000, 400,000 and 400,000 for training,validation and testing, respectively.

Baseline Methods. Five baseline methods are used in the comparison, including traditional Seq2Seq [Sutskever *et al.*, 2014], RNN autoencoder(RNN-encdec) [Cho *et al.*, 2014], Seq2Seq with attention(Seq2Seq-att) [Bahdanau *et al.*, 2015], MMI [Li *et al.*, 2016b],Back-MMI [Li *et al.*, 2016b][4] and Adver-REGS [Li *et al.*, 2017]. Since we have designed three kinds of coherence models, we have four versions of our model, denoted as Seq2SeqCo-cos, Seq2SeqCo-bi, Seq2SeqCo-MP, Seq2SeqCo-dual, respectively. We first introduce the input embeddings. For STC, we utilize character-level embeddings trained from the STC training dataset rather than word-level embeddings, due to the word sparsity, segmentation mistakes and unknown Chinese words which lead to inferior performance than character-level [Hu *et al.*, 2015]. For OSDb, we use word embeddings trained by word2vec on a large Wikipedia corpus [5]. In the training process, the dimension is set to be 300, the size of negative sample is set to be 3, and the learning rate is 0.05. Then we introduce the settings on learning parameters in the deep architecture. For a fair comparison among all the baseline methods and our methods, the numbers of hidden nodes are all set to 300, and batch sizes are set to 200. Stochastic gradient decent (SGD) is utilized in our experiment for optimization, instead of Adam, because SGD yields better performances in our experiments. The learning rate is set to be 0.5, and adaptively decays with rate 0.99 in the optimization. We run our model on a Tesla K80 GPU card with Tensorflow.

Evaluation Measures. We use both quantitative metrics and human judgements in section 4 to evaluate the proposed models. Specifically, we use two kinds of metrics for quantitative comparisons. The first one kind is traditional metrics, such as PPL and BLEU score[Xing *et al.*, 2017]. They are both widely used in natural language processing, and here we use them to evaluate the quality of the generated responses. The other kind is to evaluate the degree of diversity of the generated responses. In this paper, we use *distinct* [Li *et al.*, 2016a; 2016b], which calculates the number of distinct unigrams and bigrams in the generated responses. If a model often generates common responses, the *distinct* will be low.

6.2 Experimental Results

The quantitative evaluation results are shown in Table 2. From the results, we can see that both Back-MMI and Adver-REGS outperform traditional Seq2Seq baselines in terms of BLEU, PPL and *distinct* measures. That's because both Back-MMI and Adver-REGS further consider some reward

[3]https://github.com/jiweil/Neural-Dialogue-Generation

[4]We use the pre-trained backward seq2seq model as the reward.

[5]http://www.psych.ualberta.ca/~westburylab/downloads/westburylab.wikicorp.download.html

STC Dataset

model	BLEU	PPL	distinct-1	distinct-2
Seq2Seq	2.867	21.28	0.414	4.52
RNN-encdec	2.878	21.08	0.381	4.87
Seq2Seq-att	2.894	19.86	0.437	5.03
MMI	2.906	18.65	0.431	5.43
Back-MMI	2.917	18.57	0.441	5.82
Adver-REGS	2.918	18.57	0.433	5.74
Seq2SeqCo-cos	2.919	18.65	0.445	5.68
Seq2SeqCo-bi	2.918	18.64	0.438	5.87
Seq2SeqCo-MP	2.918	18.56	**0.448**	5.79
Seq2SeqCo-dual	**2.925**	**18.52**	0.440	**5.98**

OSDb Dataset

model	BLEU	PPL	distinct-1	disttinct-2
Seq2Seq	2.216	26.13	1.22	5.33
RNN-encdec	2.310	26.02	1.15	5.56
Seq2Seq-att	2.282	25.99	1.38	5.84
MMI	2.323	25.78	1.45	6.10
Back-MMI	2.413	25.1	1.49	6.32
Adver-REGS	2.393	25.73	1.33	6.20
Seq2SeqCo-cos	2.387	25.79	1.38	6.18
Seq2SeqCo-bi	**2.422**	25.67	1.27	6.27
Seq2SeqCo-MP	2.397	25.3	**1.52**	6.24
Seq2SeqCo-dual	**2.422**	**24.45**	1.42	**6.83**

Table 2: The metric-based evaluation results(%).

model	human score distribution(%)					Ave.	Kappa
	1	2	3	4	5		
Seq2Seq	17.4	40.2	21.7	18.5	2.2	2.48	0.454
RNN-encdec	18.7	34.1	20.9	19.8	6.6	2.62	0.517
Seq2Seq-att	14.0	35.6	14.0	31.6	4.8	2.78	0.418
MMI	13.6	30.0	12.4	39.2	4.8	2.92	0.461
Back-MMI	12.6	21.8	11.4	41.4	12.8	3.2	0.469
Adver-REGS	13.7	25.5	13.7	41.2	5.9	3.00	0.456
Seq2SeqCo-cos	11.6	28.0	9.6	42.8	8.0	3.08	0.473
Seq2SeqCo-bi	11.6	23.6	9.6	42.8	12.4	3.21	0.428
Seq2SeqCo-MP	11.2	27.2	9.2	44.0	8.4	3.11	0.426
Seq2SeqCo-dual	11.2	20.8	9.2	44.4	14.4	3.30	0.422

model	human score distribution(%)					Ave.	Kappa
	1	2	3	4	5		
Seq2Seq	10.0	54.5	21.8	9.1	4.5	2.43	0.302
RNN-encdec	8.2	51.8	18.2	16.4	5.5	2.60	0.317
Seq2Seq-att	3.6	53.6	14.5	20.0	8.2	2.75	0.286
MMI	4.5	40.0	18.2	31.8	5.5	2.94	0.285
Back-MMI	6.3	35.1	11.2	35.1	12.3	3.12	0.348
Adver-REGS	8.7	33.8	15.0	32.5	10.0	3.01	0.308
Seq2SeqCo-cos	5.5	39.1	13.6	31.8	10.0	3.02	0.341
Seq2SeqCo-bi	7.3	36.4	9.1	32.7	14.5	3.11	0.378
Seq2SeqCo-MP	4.5	38.2	12.7	33.6	10.9	3.08	0.255
Seq2SeqCo-dual	7.3	32.7	9.1	34.5	16.4	3.20	0.332

Table 3: The human evaluation on STC(top) and OSDb(bottom).

functions in the optimization process. Back-MMI uses a predefined reward function to penalize generating common responses, but the effect is quite limited. Adver-REGS uses a learned discriminator to define the reward function, though flexible, it is not clear whether the reward function truelly penalizes the common responses. Among the four proposed coherence models, the end-to-end dual learning approach performs the best, because of its capability to jointly learning all parameters for fitting the true distribution of the large scale conversation data. Our models obtain higher BLEU and lower PPL than baseline models. Take the BLEU score on STC dataset for example, the BLEU score of dual learning model is 2.925, which is significantly better than that of Back-MMI and Adver-REGS, i.e., 2.917 and 2.918. These results indicate that our models generate responses with higher quality. The distinct scores of our models are also higher than baseline models, which indicate that our models can generate more specific responses. That's because we directly penalize the case when generated probability is high and true probability is low. We also conducted the significant test, and the result shows that the improvements of our model are significant on both Chinese and English datasets, i.e., p-value < 0.01. In summary, our coherence models produce more fluent and diverse results, as compared with baseline methods.

The human evaluation results are shown in Table 3, in which the percentage of sentences belonging to each grade and the averaged grade are given to evaluate the quality of generated responses, and kappa [Fleiss, 1971] value is also given to demonstrate the consistency of different annotators. From the results, our four coherent models significantly outperform baseline methods. Take STC as an example. The averaged score of Seq2SeqCo-dual is 3.3, which is significantly better than that of Back-MMI and Adver-REGS, i.e., 3.2 and 3.0, respectively. The percentage of strongly related sentences (i.e., the sum of grade '4' and '5') of Seq2SeqCo-dual is 58.8%, which is significantly better than that of Back-MMI and Adver-REGS, i.e., 54.2% and 47.1%.

post	韦德生涯最血腥扣篮之一(The most wonderful dunk of Wade)
response	就应该补个詹姆斯的表情(Imagine the look of James)
Seq2Seq-att	这个太有才了(It is amazing)
MMI	我们的小伙伴都惊呆了(My friends were stunned)
Back-MMI	这个太有才了!(It is amazing)
Adver-REGS	我也想要这个篮球(I also want this basketball)
Seq2SeqCo-cos	韦德不是一个人(Wade is not alone)
Seq2SeqCo-bi	詹姆斯的表情太好笑了(James looks so funny)
Seq2SeqCo-MP	韦德,韦德,韦德(Wade, Wade, Wade)
Seq2SeqCo-dual	韦德的表情太有才了(Wade's expression is awesome)

post	Go ahead and kill her like you killed my son
response	You're damned if you don't kill her
Seq2Seq-att	I don't care
MMI	I don't know what you mean
Back-MMI	I don't know
Adver-REGS	What's the matter?
Seq2SeqCo-cos	I'm not a coward
Seq2SeqCo-bi	I don't care what you do with her
Seq2SeqCo-MP	I'm not going to take her out of this
Seq2SeqCo-dual	She's innocent

Table 4: The generated responses on STC(top) and OSDb(bottom).

Here we show some generated responses for demonstration. Specifically, Table 4 gives one post, its ground-truth responses, and the generated responses from different models. We can see that Seq2Seq-att, MMI, Back-MMI and Adver-REGS all produce common responses, such as *'It is amazing'*,*'I don't care'* and*'I don't know what you mean'*. However, our models give interesting responses with specific meanings. We have obtained similar observations for many other posts, but we have to omit them for space limitations.

We also computed the hit rate and hit probability as defined in our motivation. The values on the dataset of STC are 0.008449 and 0.001689 for Seq2SeqCo-dual, which are much larger than Seq2Seq-att, as described before. Therefore, our coherence models have the ability to generated responses with higher true probability.

7 Conclusion

In this paper, we first analyzed the problem of Seq2Seq in generating common responses. We find that the main reason is that the objective of Seq2Seq is equivalent to KL divergence, which does not penalize the case whose generated probability is high while the true probability is low. Then we hypothesized that the true probability can be estimated by the coherence score between post and response, inspired by our statistical findings that the similarity measure between post and response embeddings is consistent with human evaluation. Then we defined three kinds of coherence models, and used them as the reward function in a reinforcement learning framework. Experimental results on both Chinese Weibo and English Subtitle dataset showed that our models significantly outperform baselines, including traditional Seq2Seq and some recent proposed models which are related.

Acknowledgments

This work was funded by the 973 Program of China under Grant No. 2014CB340401, the National Natural Science Foundation of China (NSFC) under Grants No. 61425016, 61472401, 61722211, 61773362, and 20180290, the Youth Innovation Promotion Association CAS under Grants No. 20144310, and 2016102, and the National Key R&D Program of China under Grants No. 2016QY02D0405.

References

[Arjovsky *et al.*, 2017] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein gan. *ICML*, 2017.

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *ICLR*, 2015.

[Cho *et al.*, 2014] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *Computer Science*, 2014.

[Fleiss, 1971] Joseph L Fleiss. Measuring nominal scale agreement among many raters. *American Psychological Association*, 1971.

[Guo *et al.*, 2016] Jiafeng Guo, Yixing Fan, Qingyao Ai, and W Bruce Croft. A deep relevance matching model for ad-hoc retrieval. In *CIKM*, pages 55–64. ACM, 2016.

[Hu *et al.*, 2015] Baotian Hu, Qingcai Chen, and Fangze Zhu. Lcsts: A large scale chinese short text summarization dataset. *EMNLP*, 2015.

[Li *et al.*, 2016a] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. A diversity-promoting objective function for neural conversation models. *NAACL*, 2016.

[Li *et al.*, 2016b] Jiwei Li, Will Monroe, Alan Ritter, and Galley et al. Deep reinforcement learning for dialogue generation. *EMNLP*, 2016.

[Li *et al.*, 2017] Jiwei Li, Will Monroe, Tianlin Shi, Alan Ritter, and Dan Jurafsky. Adversarial learning for neural dialogue generation. *EMNLP*, 2017.

[Mou *et al.*, 2017] Lili Mou, Yiping Song, Rui Yan, Ge Li, Lu Zhang, and Zhi Jin. Sequence to backward and forward sequences: A content-introducing approach to generative short-text conversation. *ACL*, 2017.

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

[Shang *et al.*, 2015] Lifeng Shang, Zhengdong Lu, and Hang Li. Neural responding machine for short-text conversation. *ACL*, 2015.

[Shen *et al.*, 2017] Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, and Zhao et al. A conditional variational framework for dialog generation. *ACL*, 2017.

[Socher *et al.*, 2013] Richard Socher, Danqi Chen, Christopher D Manning, and Andrew Ng. Reasoning with neural tensor networks for knowledge base completion. In *NIPS*, pages 926–934, 2013.

[Sutskever *et al.*, 2014] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. Sequence to sequence learning with neural networks. In *NIPS*, pages 3104–3112, 2014.

[Sutton *et al.*, 2000] Richard S Sutton, David A McAllester, Satinder P Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. In *NIPS*, pages 1057–1063, 2000.

[Vinyals and Le, 2015] Oriol Vinyals and Quoc Le. A neural conversational model. *Computer Science*, 2015.

[Wan *et al.*, 2016a] Shengxian Wan, Yanyan Lan, Jiafeng Guo, Jun Xu, Liang Pang, and Xueqi Cheng. A deep architecture for semantic matching with multiple positional sentence representations. In *AAAI*, pages 2835–2841, 2016.

[Wan *et al.*, 2016b] Shengxian Wan, Yanyan Lan, Jun Xu, Jiafeng Guo, Liang Pang, and Xueqi Cheng. Match-srnn: Modeling the recursive matching structure with spatial rnn. *IJCAI*, 2016.

[Xia *et al.*, 2016] Yingce Xia, Di He, Tao Qin, Liwei Wang, Nenghai Yu, Tie-Yan Liu, and Wei-Ying Ma. Dual learning for machine translation. *NIPS*, 2016.

[Xing *et al.*, 2017] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic aware neural response generation. In *AAAI*, pages 3351–3357, 2017.

[Yu *et al.*, 2017] Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. Seqgan: Sequence generative adversarial nets with policy gradient. In *AAAI*, pages 2852–2858, 2017.