STYLEDGPT: Stylized Response Generation with Pre-trained Language Models

Ze Yang[1], **Wei Wu**[2], **Can Xu**[3], **Xinnian Liang**[1], **Jiaqi Bai**[1],
Liran Wang[1], **Wei Wang**[4], and **Zhoujun Li**[1*]
[1]State Key Lab of Software Development Environment, Beihang University, Beijing, China
[2]Meituan, Beijing, China [3]Microsoft, Beijing, China [4]China Resources Group
{tobey,xnliang,bjq,wanglr,lizj}@buaa.edu.cn
{wuwei19850318,ww.cs.tj}@gmail.com caxu@microsoft.com

Abstract

Generating responses following a desired style has great potentials to extend applications of open-domain dialogue systems, yet is refrained by lacking of parallel data for training. In this work, we explore the challenging task with pre-trained language models that have brought breakthrough to various natural language tasks. To this end, we introduce a KL loss and a style classifier to the fine-tuning step in order to steer response generation towards the target style in both a word-level and a sentence-level. Comprehensive empirical studies with two public datasets indicate that our model can significantly outperform state-of-the-art methods in terms of both style consistency and contextual coherence.

1 Introduction

With advances in neural machine learning (Sutskever et al., 2014; Gehring et al., 2017; Vaswani et al., 2017) and availability of huge amount of human conversations on social media, there has been significant progress on building open-domain dialogue systems with natural language generation techniques. Though neural generative models are notorious for replying with bland responses (Li et al., 2015), some very recent work demonstrates that response generation models learned with pre-training techniques (Radford et al., 2019) can effectively overcome the deficiency suffered by previous models and are capable of having smooth conversations with humans through reasonable and specific replies (Wolf et al., 2019; Zhang et al., 2019b).

The compelling performance exhibited by the pre-trained dialogue models encourages us to explore more difficult yet important problems in conversational AI. In this work, we study stylized response generation, that is responses provided by a model should not only be coherent with the conversation contexts, but also be consistent with a designated style. Such research could facilitate developers to customize their dialogue systems in terms of response styles, and thus broaden applications of the systems, from a social companion (Shum et al., 2018) or a virtual assistant (Ram et al., 2018) to a variety of vertical scenarios such as customer service (requiring a polite style), virtual characters in games (requiring specific personas), assistants in specific domains (requiring domain knowledge), etc. Normally, a target style is specified by a non-conversational corpus (e.g., novels, news, blogs, etc.) apart from the paired dialogue corpus (Luan et al., 2017; Niu and Bansal, 2018; Gao et al., 2019). Thus, the major challenge of the task lies in the scarcity of paired data for learning the correspondence between conversation contexts and proper responses in the desired style, which is a key factor in success of the neural dialogue models developed so far. As a result, it is very likely that a response either digresses from the context of the current dialogue (Luan et al., 2017; Gao et al., 2019), or loses fidelity to the target style (Niu and Bansal, 2018).

We consider addressing the challenge by taking advantage of the large scale pre-trained language models. The basic idea is that deep neural language models learned from huge amount of text, such as GPT-2 (Radford et al., 2019) and DialoGPT (Zhang et al., 2019b), have packed enough style knowledge into their parameters (Dathathri et al., 2020), and thus by simply steering the distribution in decoding towards the desired style, we can obtain both contextual coherence and style consistency. Following the idea, we build a response generation model on top of a pre-trained language model and devise both a word-level loss and a sentence-level loss to fine-tune the pre-trained model towards the target style. The word-level loss regularizes the likeli-

[*] Corresponding Author

Findings of the Association for Computational Linguistics: EMNLP 2020, pages 1548–1559
November 16 - 20, 2020. ©2020 Association for Computational Linguistics

hood of response generation with a KL divergence term between the probability of dialogues and the probability of stylized language estimated by fine-tuning a pre-trained language model on the style corpus, while the sentence-level loss maximizes the likelihood of a response given by the pre-trained response generation model being classified as a sentence matching the target style. We employ a Gumbel trick to overcome the obstacle in back-propagation due to the discrete nature of natural language when optimizing the sentence-level loss. The final response is selected by a sample-and-rank strategy to further enhance relevance regarding to the dialogue context and fidelity regarding to the target style.

We name our model StyleDGPT standing for "Stylized DialoGPT". Empirical studies are conducted on two tasks: arXiv-style response generation and Holmes-style response generation with the data shared in (Gao et al., 2019), where responses in the style of scientific papers and the style of Sherlock Holmes novels are pursued respectively for a given context. Besides the style intensity used in (Gao et al., 2019), we further examine style consistency from both a lexical perspective and a syntactic perspective with two new metrics. Evaluation results on both automatic metrics and human judgment indicate that our model can significantly outperform state-of-the-art methods. The code is available at `https://github.com/TobeyYang/StyleDGPT`.

Our contributions are three-fold: (1) proposal of tackling the problem of stylized response generation with pre-trained language models; (2) proposal of a word-level objective and a sentence-level objective in fine-tuning of a pre-trained language model for the task; and (3) empirical verification of the effectiveness of the proposed method on public datasets.

2 Related Work

Open-domain Dialogue Generation has received more and more attention in NLP community. Inspired by neural machine translation, early works apply the sequence-to-sequence model to this task and achieve promising results (Ritter et al., 2011; Shang et al., 2015; Vinyals and Le, 2015). Since then, various architectures have been proposed to address the key challenges in open-domain dialogue systems, including suppressing the generic responses (Li et al., 2015; Zhao et al., 2017; Xing et al., 2017a), context modeling (Serban et al., 2016, 2017; Xing et al., 2017b; Zhang et al., 2019a), controlling the attributes of responses (Xu et al., 2019; Zhou et al., 2017; Zhang et al., 2018a; Wang et al., 2018; See et al., 2019) and incorporating different types knowledge into generation (Li et al., 2016; Zhang et al., 2018b; Zhou et al., 2017; Zhao et al., 2020). In this work, we study the problem of stylized response generation, which aims to incorporate the style information from non-parallel data into the generation process.

Stylized Text Generation has attracted broad interest in recent years, especially the style transfer, which aims to alter one or more attributes of text while preserving the content. A prevalent idea of unsupervised style transfer is learning to separate "content" and "style" of text and manipulate the style to induce transfer at inference time (Li et al., 2018; Fu et al., 2018; John et al., 2019). However, some works show that the disentanglement cannot be met and is not necessary, and leverage techniques like reconstruction and back-translation introduced in unsupervised machine translation (Lample et al., 2018), transformer (Dai et al., 2019) to achieve unsupervised style transfer. Different from style transfer, stylized response generation requires that the response is coherent with its context and the content can be varied. Akama et al. (2017) first train a basic model on a large-scale dialogue corpus and then fine-tune the model with a small stylized corpus. Niu and Bansal (2018) propose three weakly-supervised methods to generate polite responses using non-parallel data. Gao et al. (2019) build a structured latent space sharing between conversation modeling and style transfer. However, limited by the sparsity of the latent space, it is difficult to balance the style and contextual coherence while sampling in the neighborhood of the latent code of context at inference time.

Pretraining Methods have led remarkable success in various NLP tasks which demonstrates its great capabilities in language understanding and text generation (Radford et al., 2018, 2019; Devlin et al., 2019; Yang et al., 2019; Liu et al., 2019; Conneau and Lample, 2019; Clark et al., 2020). Recently, the pretraining methods have also been used to tackle the key challenges in dialogue systems such as context representation (Mehri et al., 2019), response selection (Henderson and Su, 2019), knowledge-grounded response generation (Zhao et al., 2020) and personalized response generation (Zheng et al., 2019). In particular, the large-scale pre-trained open-domain dialogue systems (Zhang et al., 2019b; Adiwardana et al., 2020) make a large step towards human-like chatbot against previous works which rely on complex frameworks developed over many years. On this basis, we propose to study the open-domain stylized response generation with pre-trained models in this work.

3 Problem Formalization

Suppose that we have a dialogue corpus $\mathcal{D}_{conv} = \{(X_i, Y_i)\}_{i=1}^{n}$ and a style corpus $\mathcal{D}_{style} = \{S_i\}_{i=1}^{m}$, where $\forall (X_i, Y_i) \in \mathcal{D}_{conv}$, X_i is a conversation context and Y_i a response to X_i, and $\forall S_i \in \mathcal{D}_{style}$, S_i is a piece of text in the target style $\mathcal{S}$. We do not assume that there exists pairs $\{(X, Y')\}$ with Y' expressed in the style $\mathcal{S}$[1], and $\mathcal{D}_{style}$ could be collected from text in an arbitrary style (e.g. scientific papers, novels, etc.). Our goal is to learn a generation model $P(Y|X, \mathcal{S})$ with both $\mathcal{D}_{conv}$ and $\mathcal{D}_{style}$, and thus given a new context X, one can generate a response Y that properly replies to the context X following the style $\mathcal{S}$.

4 Approach

We employ DialoGPT (Zhang et al., 2019b) as the general response generation model $P(Y|X)$, and try to bias $P(Y|X)$ towards the language distribution $P(S)$ estimated from $\mathcal{D}_{style}$ in fine-tuning. Below, we first briefly review the OpenAI GPT-2 (Radford et al., 2019) and DialoGPT, which serve as the backbone of our model. Then, we introduce two learning objectives from both a word perspective and a sentence perspective to interpolate style $\mathcal{S}$ into response generation.

4.1 Backbone Networks

GPT-2 is a large transformer based generative model pre-trained with language modeling (Radford et al., 2019). Given a sequence $X = (x_0, \cdots, x_n)$, the generative probability $p(X)$ can be factorized as the product of conditional probabilities over the tokens (Jelinek, 1980; Bengio et al., 2003):

$$p(X) = p(x_0) \prod_{i=1}^{n} p(x_i|x_0, \cdots, x_{i-1}) \quad (1)$$

GPT-2 uses a multi-layer transformer to model the distributions in a recurrent way. At step t, let us define $\mathbf{H}_t = [(\mathbf{K}_t^{(1)}, \mathbf{V}_t^{(1)}), \cdots, (\mathbf{K}_t^{(l)}, \mathbf{V}_t^{(l)})]$ as the past key-value matrices where $(\mathbf{K}_t^{(i)}, \mathbf{V}_t^{(i)})$ represents the key-value pairs computed by the i-th layer from step 0 to step t, then given the input token x_t, the distribution of the next token x_{t+1} can be efficiently calculated using the cached $\mathbf{H}_t$ which is formulated as:

$$\begin{aligned} &e_{x_t} = \mathbf{E}\, x_t^*, \\ &o_{x_{t+1}}, \mathbf{H}_{t+1} = \text{Transformer}(e_{x_t}, \mathbf{H}_t), \\ &p(x_{t+1}|x_0, \cdots, x_t) = \text{softmax}(\mathbf{W}_o\, o_{x_{t+1}}), \end{aligned} \quad (2)$$

where $\mathbf{E} \in \mathbb{R}^{d_e \times |V|}$ is the word embedding matrix with d_e the dimension and $|V|$ the vocabulary size, $x_t^* \in \mathbb{R}^{|V|}$ is a one-hot vector corresponding to token x_t, $o_{x_{t+1}} \in \mathbb{R}^{d_c}$ is the hidden state at step t with d_c the hidden size, and $\mathbf{W}_o \in \mathbb{R}^{|V| \times d_c}$ is a parameter matrix that maps the hidden state $o_{x_{t+1}}$ to a logit vector in the size of $|V|$. At inference time, x_{t+1} is predicted following $p(x_{t+1}|x_0, \cdots, x_t)$. Moreover, GPT-2 can also be used for language understanding. In this scenario, $o_X = (o_{x_1}, \cdots, o_{x_{n+1}})$ are treated as the representations of sequence X.

DialoGPT is a large conversational response generation model trained on 147M conversation-like exchanges from Reddit community (Zhang et al., 2019b). It inherits from GPT-2 and frames the response generation task as language modeling. For a context-response pair (X, Y), a special token $\langle|\texttt{endoftext}|\rangle$ is appended at the end of each dialogue turn and then all turns are concatenated into a long sequence. Let M denote the length of the context sub-sequence and $(x_0, \cdots, x_{M-1}, \cdots, x_N)$ denote the dialogue sequence after concatenation, the conditional generation probability of response Y is defined as:

$$p(Y|X) = \prod_{i=M}^{N} p(x_i|x_0, \cdots, x_{i-1}). \quad (3)$$

4.2 Response Style Controlling

Word-Level Objective encourages the pre-trained response generation model $P(Y|X)$ (i.e. DialoGPT) to pick words expressing the desired style $\mathcal{S}$ in decoding. Specifically, we train a language model $P(S)$ with $\mathcal{D}_{style}$ on the basis of GPT-2 and use it as regularization to drive $P(Y|X)$ towards $P(S)$. It is inspired that if a response Y is not consistent with the style $\mathcal{S}$, it will get high

[1] Some pairs in $\mathcal{D}_{conv}$ may meet the condition, but there is not an oracle that can tell us the information.

perplexity (i.e. Y is far from the language space of $\mathcal{S}$). Furthermore, $P(S)$ could not only provide an overall evaluation on the fidelity of a response Y, but also assign a direct probability distribution over the vocabulary at each step and thus provide word-level information about which words need to be promoted in generation.

For each $(X, Y) \in \mathcal{D}_{conv}$, we denote $p_Y = (p_{y_1}, \cdots, p_{y_m})$ (m is the length of Y) as the next-word distributions of Y given by $P(Y|X)$. Meanwhile, we feed Y into $P(S)$ and obtain the next-word distributions $\hat{p}_Y = (\hat{p}_{y_1}, \cdots, \hat{p}_{y_m})$. Then the word-level objective is formulated as:

$$\mathcal{L}_w = \mathbb{E}_{(X,Y)\sim\mathcal{D}_{conv}}\, d(p_Y \| \, \hat{p}_Y), \tag{4}$$

where $d(p_Y \| \, \hat{p}_Y)$ could be any metrics measuring the *distance* between p_Y and $\hat{p}_Y$. Here, we specify $d(\cdot \| \cdot)$ as the Kullback-Leibler (KL) divergence. Then, $d(p_Y \| \, \hat{p}_Y) = \sum_{i=1}^{m} \mathrm{D}_{KL}(p_{y_i} \| \, \hat{p}_{y_i})$. At each step, $\mathcal{L}_w$ modifies the next-word distribution in the direction of $P(\mathcal{S})$ where the probabilities of words with the desired style $\mathcal{S}$ will be increased, which can encourage the selection of these words at inference time.

Sentence-level Objective modifies $P(Y|X)$ towards the target style $\mathcal{S}$ from a syntactic and semantic perspective. In training, we hope that a response matching style $\mathcal{S}$ could have more impact in guiding the optimization of $P(Y|X)$ towards the desired direction. To this end, we first train a discriminative model $P(\mathcal{S}|X)$ to predict whether the input sequence X matches the style $\mathcal{S}$. Formally, given an input sequence $X = (x_0, \cdots, x_n)$, the probability is defined as:

$$\begin{aligned} p(\mathcal{S}|X) &= \mathrm{sigmoid}(\mathbf{W}_d\, \hat{o}_X), \\ \hat{o}_X &= \mathrm{average\_pooling}(o_X), \end{aligned} \tag{5}$$

where $o_X = (o_{x_1}, \cdots, o_{x_{n+1}})$ are the representations of X encoded by GPT-2, average_pooling$(\cdot)$ denotes the average pooling layer where the i-th element $\hat{o}_X^{(i)}$ is given by $\frac{1}{n+1}\sum_{j=1}^{n+1} o_{x_j}^{(i)}, i \in [1, d_c]$, and $\mathbf{W}_d \in \mathbb{R}^{1\times d_c}$ is a parameter. In the training phase, positive examples are sampled from $\mathcal{D}_{style}$ while negative examples are utterances sampled from $\mathcal{D}_{conv}$ [2]. Then the sentence-level objective is formulated as:

$$\mathcal{L}_s = \mathbb{E}_{(X,Y)\sim\mathcal{D}_{conv}, \widetilde{Y}\sim P(\widetilde{Y}|X)}[-\log p(\mathcal{S}|\widetilde{Y})]. \tag{6}$$

$\mathcal{L}_s$ aims to regularize the output of the generation model by ascending the probability given by the discriminative model $P(\mathcal{S}|X)$, which is similar to the optimization process of the generator in GANs (Goodfellow et al., 2014). The challenge is that since $\widetilde{Y}$ is discrete, it is impossible to back-propagate through sampling from $P(\widetilde{Y}|X)$. Although it can be circumvented by using the reinforcement learning (RL) algorithm (Sutton et al., 2000), the performance is not satisfactory in our experiments. In this work, we propose using the Gumbel trick (Jang et al., 2016) to tackle the challenge. At step t, instead of sampling a token from $p(x_{t+1}|x_0, \cdots, x_t)$, the input vector of step $t+1$ is obtained by:

$$x_{t+1}^{*} = \mathrm{gumbel\_softmax}(\mathbf{W_o}\, o_t, \tau), \tag{7}$$

where τ is the temperature and when $\tau \to 0$, $x_{t+1}^{*} \in \mathbb{R}^{|V|}$ becomes a one-hot vector.

Training Objective. The two objectives presented above are able to drive $P(Y|X)$ to generate responses with desirable style $\mathcal{S}$, but it will quickly result in irrelevant responses as both of them only focus on responses. To overcome this, we preserve the negative log-likelihood (NLL) loss in DialoGPT to maintain the relevance between the context and response:

$$\mathcal{L}_{NLL} = \mathbb{E}_{(X,Y)\sim\mathcal{D}_{conv}}[-\log p(Y|X)] \tag{8}$$

The final training loss is the weighted sum of the word-level loss, sentence-level loss, and relevance loss:

$$\mathcal{L} = \lambda_w \cdot \mathcal{L}_w + \lambda_s \cdot \mathcal{L}_s + \lambda_{NLL} \cdot \mathcal{L}_{NLL}, \tag{9}$$

where λ_w, λ_s, λ_{NLL} are three weight scalars.

Sampling and Ranking. Because it is possible to generate non-stylized responses at inference time, we employ the sample-and-rank decoding strategy following Gao et al. (2019). First, we sample N independent candidate responses for each context by using top-k sampling method with temperature T. Then, we re-rank them in terms of both relevance and style intensity and select the candidate with the highest score as the final response. The score of a candidate Y_i for context X is defined as

$$score(Y_i) = \beta \cdot p(Y_i|X) + (1-\beta) \cdot p(\mathcal{S}|Y_i), \tag{10}$$

where $p(Y_i|X)$ measures relevance of Y_i regarding to X, $p(\mathcal{S}|Y_i)$ returns style intensity of Y_i defined

[2] The ratio of the positive and the negative is $1 : 5$ in our experiments.

by the discriminative model $P(\mathcal{S}|X)$, and β is a hyper-parameter.

5 Experiments

5.1 Datasets

In order to verify the effectiveness of our model, we experiment on two tasks: generating arXiv-style and Holmes-style responses. The statistics of datasets are summarized in Table 1. The datasets are constructed following the pipeline in Gao et al. (2019). The style corpus $\mathcal{D}_{style}$ for arXiv-style response generation task consists of ~1M sentences that are extracted from the LaTex source code of papers on website arXiv.org from 1998 to 2002 [3]. For Holmes-style response generation task, $\mathcal{D}_{style}$ contains ~38k sentences built from ebooks of Sherlock Holmes novel series downloaded from the site Gutenberg.org [4]. Both tasks share the same conversation dataset $\mathcal{D}_{conv}$ which consists of 10M context-response pairs extracted from user posts and comments on site Reddit.com during the year 2011 [5]. The validation set $\mathcal{D}_{val}$ and the test set $\mathcal{D}_{test}$ are constructed by filtering the Reddit data in 2013 with the classifier in (Gao et al., 2019) (intensity score > 0.4) [6]. As Gao et al. (2019) do not release their test data, nor specify the size of the test set, we randomly select 2k/2k samples as the validation/test sets, and each context has at least 4 responses.

Task	Training		Validation	Test
	$\mathcal{D}_{conv}$	$\mathcal{D}_{style}$	$\mathcal{D}_{val}$	$\mathcal{D}_{test}$
arXiv-style	Reddit 10,000,000	arXiv 1,347,538	arXiv-style 2,000	Reddit 2,000
Holmes-style	Reddit 10,000,000	Holmes 38,309	Holmes-style 2,000	Reddit 2,000

Table 1: Tasks and datasets

5.2 Evaluation Methodology

We compare different models with both automatic metrics and human judgment.

Automatic Metrics. For automatic evaluation, we measure the quality of generated responses from three aspects: **Style Consistency**, **Relevance**, and **Diversity**. The relevance is measured with BLEU (Papineni et al., 2002) and Rouge (Lin, 2004) [7]. To evaluate diversity, we follow Li et al. (2015) and use Distinct-1 (Dist-1) and Distinct-2 (Dist-2) as metrics which are calculated as ratios of distinct unigrams and bigrams in responses, respectively.

In terms of style consistency, existing work only measures the style intensity using classifiers (Gao et al., 2019). However, the style of text is an amalgam, and differences between two styles are reflected in multiple linguistic dimensions (Verma and Srinivasan, 2019). Thus, we propose to evaluate the style of response from three perspectives: (1) ***Intensity***: we report the scores from the discriminative model $p(\mathcal{S}|X)$[8]. (2) ***Lexical***: it is a word-level metric that measures the distance between two lexical distributions. We first build a lexicon with all the ngrams ($N = 1, 2, 3, 4$) from $\mathcal{D}_{conv}$ and $\mathcal{D}_{style}$ (i.e., Reddit, arXiv, and Holmes corpora). To reduce noise, ngrams that occur less than 10 times are filtered out and there are $1,346,175$ distinct ngrams left. Then the lexical distributions of a model and the target style can be represented as normalized $1,346,175$-dimensional vectors with each element the frequency of the corresponding ngram in the generated responses (over the test set) and $\mathcal{D}_{style}$ respectively. Finally, we calculate the Jensen-Shannon divergence (Fuglede and Topsoe, 2004) to measure the distance of the two vectors. (3) ***Syntactic***: it is a sentence-level metric. Motivated by Feng et al. (2012), the style of text can be recognized by the ratio of the following 5 syntactic types: (a) *simple*; (b) *compound*; (c) *complex*; (d) *complex-compound*; (e) *others*. The type of a sentence is determined by the algorithm proposed by Feng et al. (2012) which relies on the PCFG tree parsed by the Stanford CoreNLP [9]. We compute the distributions of the style corpus and responses generated by models and report the Jensen-Shannon divergence.

Human Evaluation. We recruit 3 well-educated native speakers as annotators to compare our model with each of the baselines. Each annotator checks one context with two responses at a time with one response from our model and the other from a base-

[3] downloaded from http://www.cs.cornell.edu/projects/kddcup/datasets.html

[4] http://www.gutenberg.org

[5] We use the raw data collected by a third party http://files.pushshift.io/reddit.

[6] available at https://github.com/golsun/StyleFusion/tree/master/classifier

[7] Both metrics are computed by scripts of a public NLG evaluation project available at https://github.com/Maluuba/nlg-eval.

[8] The evaluation is more accurate than that from the classifiers available at https://github.com/golsun/StyleFusion/tree/master/classifier because of the capability of GPT-2.

[9] https://stanfordnlp.github.io/CoreNLP

Models	Style Consistency			Relevance (↑)			Diversity (↑)	
	Intensity (↑)	Lexical (↓)	Syntactic (↓)	BLEU1	BLEU2	RougeL	Dist-1	Dist-2
arXiv-style Response Generation								
MTask (Luan et al., 2017)	0.284	0.7565	0.2653	13.42	3.56	11.53	0.040	0.091
S2S+LM (Niu and Bansal, 2018)	0.399	0.7484	0.2549	15.25	4.62	10.41	0.052	0.273
StyleFusion (Gao et al., 2019)	0.412	0.7582	0.2282	16.81	5.69	10.82	0.055	0.107
DialoGPT (Zhang et al., 2019b)	0.208	0.6518	0.2561	17.84	5.20	10.68	**0.296**	**0.711**
StyleDGPT	**0.503**	**0.6237**	**0.1912**	**19.04**	**5.74**	**12.49**	0.228	0.614
Holmes-style Response Generation								
MTask (Luan et al., 2017)	0.276	0.7106	0.2356	24.47	8.87	16.03	0.027	0.063
S2S+LM (Niu and Bansal, 2018)	0.450	0.5982	0.1959	25.32	9.15	14.82	0.051	0.304
StyleFusion (Gao et al., 2019)	0.479	0.7023	0.1946	25.91	9.68	15.87	0.045	0.098
DialoGPT (Zhang et al., 2019b)	0.282	0.5814	0.1598	27.19	8.31	14.78	**0.172**	**0.589**
StyleDGPT	**0.602**	**0.4807**	**0.0861**	**29.58**	**10.15**	**17.10**	0.101	0.452

Table 2: Evaluation results on automatic metrics. Numbers in **bold** indicate the best performing models under the corresponding metrics. ↑/↓ means higher/lower values are better, respectively. The unit for relevance is percentage.

line model, and the two responses are shown in random order. The annotators then are asked to compare them on four aspects: (1) **Style Consistency**: if the response exhibits the desired style $\mathcal{S}$; (2) **Fluency**: if the response is fluent without any grammatical errors; (3) **Relevance**: if the response is coherent with the given context; and (4) **Informativeness**: if the response is rich in content and thus could keep the conversation going. For each aspect, if the annotator cannot tell which response is better, he/she is asked to label a “Tie”. For each task, 200 test examples are sampled for annotation. Each pair of responses receive 3 labels on each of the three aspects, and the agreement among the annotators are measured by Fleiss’ kappa (Fleiss and Cohen, 1973).

5.3 Baselines

We compare our model with the following baselines: (1) **MTask**: a vanilla multi-task learning model proposed by Luan et al. (2017) trained with both $\mathcal{D}_{conv}$ and $\mathcal{D}_{style}$. We use the code implemented by Gao et al. (2019) included in the project `https://github.com/golsun/StyleFusion`. (2) **S2S+LM**: the fusion model proposed by Niu and Bansal (2018) that merges the decoder of a seq2seq model trained on $\mathcal{D}_{conv}$ and a language model trained on $\mathcal{D}_{style}$ by weighted averaging the word distributions at inference time. We use the code published at `https://github.com/WolfNiu/polite-dialogue-generation`. (3) **StyleFusion**: the regularized multi-task learning model proposed by Gao et al. (2019) which builds a structured latent space to bridge the conversation modeling and style transfer. The model is jointly learned with $\mathcal{D}_{conv}$ and $\mathcal{D}_{style}$. We run the code released at `https://github.com/golsun/StyleFusion` with default settings. (4) **DialoGPT**: an open-domain pre-trained response generation model built upon GPT-2 that attains a performance close to human (Zhang et al., 2019b). We use the 345M fine-tuned model which can be downloaded from `https://github.com/microsoft/DialoGPT`.

5.4 Implementation Details

Our models are implemented with the Huggingface transformers repository [10]. To balance cost and effect, the language model $P(\mathcal{S})$ and the discriminative model $P(\mathcal{S}|X)$ are built upon GPT-2 (117M) with 12 layers and 768 hidden units. The embedding layer and the transformer module are shared between two models, and we only optimize the parameters of the projection layer and the classification layer, respectively. We choose DialoGPT (345M) as the basis of StyleDGPT which has 24 layers and 1024 hidden units. In both tasks, we use the vocabulary published along with GPT-2 by OpenAI that contains $50,257$ tokens. The temperature τ of gumabel softmax is set as 0.1. Hyper-parameters are selected via grid search, and $\lambda_w/\lambda_s/\lambda_r$ are finally set as $0.0005/0.05/1$ for the arXiv-style response generation task and $0.005/0.05/1$ for the Holmes-style response generation task, respectively. All models are trained with the Adam optimizer (Kingma and Ba, 2015) ($\beta_1 = 0.9$, $\beta_2 = 0.999$) with a learning rate of 5×10^{-7}. We choose $k = 40$ and $T = 1.0$ in top-k decoding following (Radford et al., 2019; Adiwardana et al., 2020). At inference time, all approaches including our model and baselines generate 50 candidates for each context (i.e. $N = 50$), and the top one candidate is selected for evaluation

[10] `https://github.com/huggingface/transformers`

Models	Style Consistency			Fluency			Relevance			Informativeness			Kappa
	W(%)	L(%)	T(%)	W(%)	L(%)	T(%)	W(%)	L(%)	T(%)	W(%)	L(%)	T(%)	
arXiv-style Response Generation													
StyleDGPT vs. MTask	43.6	25.2	31.2	25.5	20.0	54.5	31.3	20.5	48.2	37.4	20.0	43.6	0.62
StyleDGPT vs. S2S+LM	41.7	21.6	36.7	39.0	7.8	53.2	53.3	10.3	36.4	38.2	17.3	44.5	0.67
StyleDGPT vs. StyleFusion	38.2	18.4	43.4	23.6	18.3	58.1	38.0	16.2	45.8	31.8	15.2	53.0	0.65
StyleDGPT vs. DialoGPT	51.3	10.2	38.5	16.2	21.8	62.0	21.2	26.5	52.3	23.2	23.8	53.0	0.61
Holmes-style Response Generation													
StyleDGPT vs. MTask	46.3	13.8	39.1	28.0	14.8	57.2	43.8	15.4	40.8	36.8	12.0	51.2	0.65
StyleDGPT vs. S2S+LM	45.0	19.5	35.5	36.3	4.8	58.9	52.2	9.0	38.8	38.6	16.3	45.1	0.61
StyleDGPT vs. StyleFusion	36.2	18.0	45.8	31.4	11.5	57.1	36.0	17.5	46.5	41.3	12.2	46.5	0.70
StyleDGPT vs. DialoGPT	52.0	13.3	34.7	14.4	12.6	73.0	19.3	20.5	60.2	22.6	15.8	61.6	0.63

Table 3: Human annotation results. W, L, and T refer to Win, Lose, and Tie, respectively. The ratios are calculated by combining labels from the three annotators.

Models	Style Consistency			Relevance (↑)			Diversity (↑)	
	Intensity (↑)	Lexical (↓)	Syntactic (↓)	BLEU1	BLEU2	RougeL	Dist-1	Dist-2
arXiv-style Response Generation								
StyleDGPT	0.503	0.6237	0.1912	19.04	5.74	12.49	0.228	0.614
StyleDGPT (w/o $\mathcal{L}_w$)	0.378	0.6357	0.2165	18.66	5.69	11.84	0.260	0.651
StyleDGPT (w/o $\mathcal{L}_s$)	0.670	0.6213	0.2177	17.28	4.85	11.39	0.182	0.564
StyleDGPT (w/o $\mathcal{L}_{NLL}$)	0.880	0.5712	0.1594	13.16	4.08	11.86	0.046	0.273
Holmes-style Response Generation								
StyleDGPT	0.602	0.4807	0.0861	29.58	10.15	17.10	0.101	0.452
StyleDGPT (w/o $\mathcal{L}_w$)	0.497	0.5007	0.1194	29.21	9.34	16.14	0.130	0.514
StyleDGPT (w/o $\mathcal{L}_s$)	0.680	0.4716	0.1551	27.89	9.22	16.54	0.097	0.459
StyleDGPT (w/o $\mathcal{L}_{NLL}$)	0.891	0.4709	0.1521	26.54	8.56	15.53	0.049	0.298

Table 4: Ablation results on automatic metrics.

according to Equation (10).

5.5 Evaluation Results

Automatic Evaluation. Table 2 reports the evaluation results on automatic metrics. Without any complicated manipulation on latent spaces, StyleDGPT outperforms the non-pre-trained baselines with large margins on all metrics in both tasks, demonstrating the advantage of pre-training over the state-of-the-art method in stylized response generation. The significant improvement over the vanilla DialoGPT on style consistency indicates that StyleDGPT can effectively leverage the extra objectives and bias response decoding towards the desired style. Moreover, it seems that forcing responses to a particular style (i.e., arXiv style and Holmes style) is also helpful in relevance, though there is a sacrifice on diversity. This is because the search space in decoding now becomes more concentrated on words that can express the target styles[11].

Human Evaluation. Table 3 reports the results of human evaluation. The values of kappa are all above 0.6, indicating substantial agreement among the three annotators. We can see StyleDGPT outperforms all non-pre-trained baselines on the three aspects, which echoes the results of automatic evaluation. Specifically, S2S+LM achieves poor performance on fluency because the weighted average of the token distributions predicted by the language model and the seq2seq decoder harms their attributes of language modeling, which also leads to low relevance. Compared to DialoGPT, we notice that StyleDGPT significantly improves upon style consistency while achieves comparable performance on relevance and informativeness, which demonstrates the effectiveness of the proposed objectives in fine-tuning.

5.6 Discussions

Ablation Study. To understand the roles of $\mathcal{L}_w$, $\mathcal{L}_s$, and $\mathcal{L}_{NLL}$ in learning to generate stylized responses, we remove them one at a time from the full objective in Equation (9), and then check the performance of the variants of StyleDGPT on the test sets. Table 4 reports the evaluation results. We can see that (1) all the three objectives are useful, as removing any of them will cause a performance drop on some metrics; (2) $\mathcal{L}_w$ is more important to lexical consistency while $\mathcal{L}_s$ is more important to syntactic consistency, which echoes our motivation in design of the two objectives; and (3) without $\mathcal{L}_{NLL}$, the model will be misled by the style corpus

[11] Note that human responses for calculating the relevance metrics are biased to the target styles according to a style classifier.

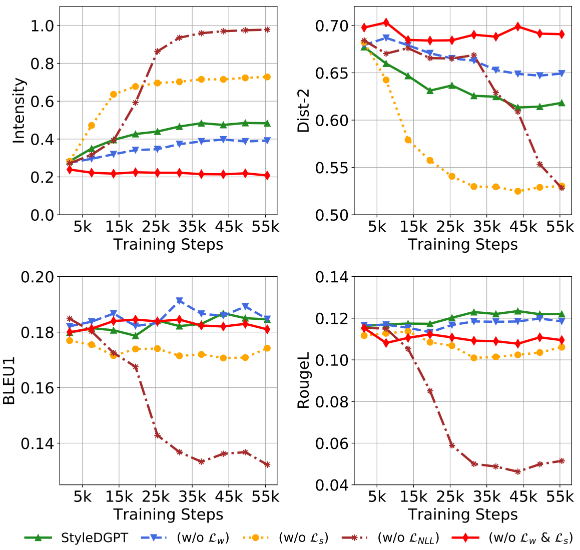

Figure 1: Trajectories of ablated STYLEDGPT on the validation set of arXiv-style response generation.

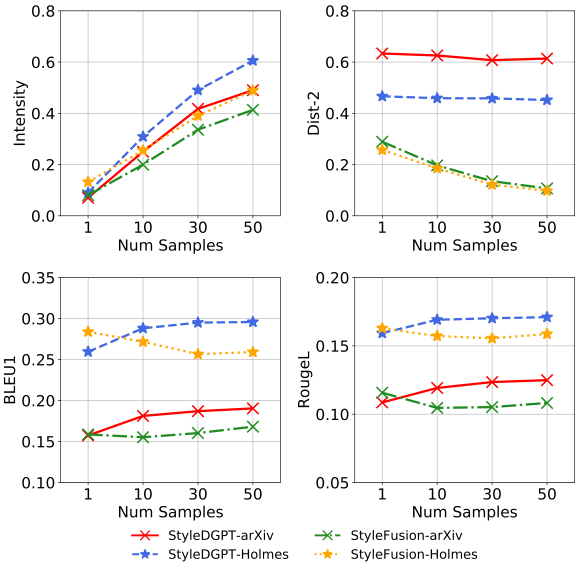

Figure 2: Comparisons over the number of sampled candidates on both tasks.

and lose the connection with conversation contexts.

Since $\mathcal{L}_w$, $\mathcal{L}_s$, and $\mathcal{L}_{NLL}$ are coordinated in learning of STYLEDGPT, more insights about the effect of the objectives can be obtained by checking the trajectories of the variants on validation, as illustrated by Figure 1[12]. Without $\mathcal{L}_s$, there is a steady and significant improvement on style intensity but dramatic drops on BLEU1, RougeL, and Dist-2 (compared with the model without both $\mathcal{L}_s$ and $\mathcal{L}_w$), which indicates that $\mathcal{L}_w$ can provide stronger guidance regarding style expression than $\mathcal{L}_s$. On the other hand, comparing STYLEDGPT w/o $\mathcal{L}_w$ and STYLEDGPT w/o $\mathcal{L}_w$ & $\mathcal{L}_s$, we find that $\mathcal{L}_s$ can gradually and moderately improve upon style intensity and relevance with only a little hurt on diversity. Finally, when $\mathcal{L}_{NLL}$ is removed, the model will quickly forget conversation contexts and converge to the style language model. The full model balances the effect of the three losses and attains both style consistency and contextual coherence, though it has to suffer from diversity drop due to the existence of $\mathcal{L}_w$.

Impact of the Sampling Number N. To understand how the sample-and-rank strategy affects model performance, we evaluate STYLEDGPT and StyleFusion by varying the sampling number N in $\{1, 10, 30, 50\}$ on both tasks. Figure 2 shows the results. We observe that (1) style intensity is more sensitive to the value of N than other metrics; (2) though the two models are comparable in terms of style intensity when $N = 1$, STYLEDGPT can exhibit the desired styles with fewer samples; (3) STYLEDGPT is always better than StyleFusion on Dist-2, thanks to DialoGPT; and (4) while STYLEDGPT is able to attain both style consistency and contextual coherence with enough samples, it is difficult for StyleFusion to balance the two aspects, as when N increases, both BLEU1 and RougeL drop. This is because when sampling in the neighborhood of the representation of a context in the structured latent space, reaching a stylized but less relevant point becomes easier when the number of samples increases.

Case Study. Finally, we conduct qualitative analysis with some examples given in Table 5 and Table 6. First, we find that the non-pre-trained models can generate interesting responses occasionally (e.g., “the ring of fire” and “the first harry potter movie.” in Table 6), but they are not always coherent with the contexts nor specific. In contrast, responses from the pre-trained models (i.e., DialoGPT and STYLEDGPT) are highly coherent with the contexts and rich in content. In terms of style consistency, responses given by STYLEDGPT are more stylized that contain more words specific to a certain style. For example, “constant state” and “quantum fluctuations” are concepts in physical science, “the behavior” of “species” is the research object of ethology, and “accident”, “investigate”, “discover” and “trap” are topical words in the Sherlock Holmes novel series.

[12] Similar trends are observed on Holmes-style response generation.

Context	We've reached the peak of human intelligence.
MTask	i think you're missing the point.
S2S+LM	i think that's a pretty good idea.
StyleFusion	i don't know what's going on.
DialoGPT	It is going to get a little higher and higher every couple of months.
STYLE-DGPT	so what happens once we reach a constant state of quantum fluctuations?
Human	This is like pure consciousness thinking of consciousness, the rational endpoint of the enlightenment dialectic.
Context	My rabbit does this every day.. not sure what he's thinking about.
MTask	i'm not sure what is going on here.
S2S+LM	i think that's a cute cat.
StyleFusion	i'm guessing he's a male or female.
DialoGPT	He was thinking of lubing his ears.
STYLE-DGPT	This is a very good point and i also would like to discuss some other differences in the behavior of this species.
Human	some sort of inner monologue.

Table 5: The example outputs on the arXiv-style response generation task.

Context	What is your favorite short story ?
MTask	the first time i saw this i thought it was a good idea to put a ring on it.
S2S+LM	the ending is a good thing and the most awesome thing i 've done
StyleFusion	the ring of fire
DialoGPT	The Last Question by David Foster Wallace
STYLE-DGPT	The one about the day before the accident, about an old lady who used to live in a shack.
Human	The multitude of short stories that my wife writes
Context	If your username was a movie, what would be the plot?
MTask	the first harry potter movie.
S2S+LM	there's a lot of things about the movie.
StyleFusion	it's a trap!
DialoGPT	Probably The Prestige
STYLE-DGPT	a story of a mad scientist who goes in to investigate something and discovers he's trapped in a cave
Human	two lovers escape to the great northwest

Table 6: The example outputs on the Holmes-style response generation task.

6 Conclusions

We exploit the pre-trained language models on the stylized response generation task. To incorporate the style information from the non-parallel data into the generation model, we propose two learning objectives from word level and sentence level to steer the output distribution towards the desired style. Evaluation results on arXiv-style and Holmes-style response generation tasks indicate the effectiveness of the proposed approach.

Acknowledgment

This work was supported in part by the National Natural Science Foundation of China (Grant Nos.U1636211, 61672081, 61370126), the Beijing Advanced Innovation Center for Imaging Technology (Grant No.BAICIT2016001), and the Fund of the State Key Laboratory of Software Development Environment (Grant No.SKLSDE2019ZX-17).

References

Daniel Adiwardana, Minh-Thang Luong, David R So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, et al. 2020. Towards a human-like open-domain chatbot. *arXiv preprint arXiv:2001.09977*.

Reina Akama, Kazuaki Inada, Naoya Inoue, Sosuke Kobayashi, and Kentaro Inui. 2017. Generating stylistically consistent dialog responses with transfer learning. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 408–412, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Yoshua Bengio, Réjean Ducharme, Pascal Vincent, and Christian Jauvin. 2003. A neural probabilistic language model. *Journal of machine learning research*, 3(Feb):1137–1155.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *Advances in Neural Information Processing Systems*, pages 7057–7067.

Ning Dai, Jianze Liang, Xipeng Qiu, and Xuanjing Huang. 2019. Style transformer: Unpaired text style transfer without disentangled latent representation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5997–6007, Florence, Italy. Association for Computational Linguistics.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

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

Technologies, Volume 1 (Long and Short Papers), pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Song Feng, Ritwik Banerjee, and Yejin Choi. 2012. Characterizing stylistic elements in syntactic structure. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1522–1533, Jeju Island, Korea. Association for Computational Linguistics.

Joseph L Fleiss and Jacob Cohen. 1973. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and psychological measurement*, 33(3):613–619.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation.

B. Fuglede and F. Topsoe. 2004. Jensen-shannon divergence and hilbert space embedding. In *International Symposium onInformation Theory, 2004. ISIT 2004. Proceedings.*, pages 31–.

Xiang Gao, Yizhe Zhang, Sungjin Lee, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2019. Structuring latent spaces for stylized response generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1814–1823, Hong Kong, China. Association for Computational Linguistics.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1243–1252. JMLR. org.

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.

Matthew Henderson and Pei-Hao Su. 2019. Convert: Efficient and accurate conversational representations from transformers. *arXiv preprint arXiv:1911.03688*.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Frederick Jelinek. 1980. Interpolated estimation of markov source parameters from sparse data. In *Proc. Workshop on Pattern Recognition in Practice, 1980*.

Vineet John, Lili Mou, Hareesh Bahuleyan, and Olga Vechtomova. 2019. Disentangled representation learning for non-parallel text style transfer. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 424–434, Florence, Italy. Association for Computational Linguistics.

Diederik P Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *ICLR*.

Guillaume Lample, Sandeep Subramanian, Eric Smith, Ludovic Denoyer, Marc'Aurelio Ranzato, and Y-Lan Boureau. 2018. Multiple-attribute text rewriting.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. 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*, pages 110–119.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016. A persona-based neural conversation model. In *ACL*, pages 994–1003.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: a simple approach to sentiment and style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1865–1874, New Orleans, Louisiana. Association for Computational Linguistics.

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

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach.

Yi Luan, Chris Brockett, Bill Dolan, Jianfeng Gao, and Michel Galley. 2017. Multi-task learning for speaker-role adaptation in neural conversation models. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 605–614, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Shikib Mehri, Evgeniia Razumovskaia, Tiancheng Zhao, and Maxine Eskenazi. 2019. Pretraining methods for dialog context representation learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3836–3845, Florence, Italy. Association for Computational Linguistics.

Tong Niu and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9.

Ashwin Ram, Rohit Prasad, Chandra Khatri, Anu Venkatesh, Raefer Gabriel, Qing Liu, Jeff Nunn, Behnam Hedayatnia, Ming Cheng, Ashish Nagar, et al. 2018. Conversational ai: The science behind the alexa prize. *arXiv preprint arXiv:1801.03604*.

Alan Ritter, Colin Cherry, and William B Dolan. 2011. Data-driven response generation in social media. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pages 583–593.

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? how controllable attributes affect human judgments. *arXiv preprint arXiv:1902.08654*.

Iulian Vlad Serban, Alessandro Sordoni, Yoshua Bengio, Aaron C Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *AAAI*, volume 16, pages 3776–3784.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *ACL*, pages 1577–1586.

Heung-Yeung Shum, Xiaodong He, and Di Li. 2018. From eliza to xiaoice: Challenges and opportunities with social chatbots. *Frontiers of IT & EE*, 19(1):10–26.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.

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.

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.

Gaurav Verma and Balaji Vasan Srinivasan. 2019. A lexical, syntactic, and semantic perspective for understanding style in text. *arXiv preprint arXiv:1909.08349*.

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 *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2193–2203.

Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.

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

Chen Xing, Wei Wu, Yu Wu, Ming Zhou, Yalou Huang, and Wei-Ying Ma. 2017b. Hierarchical recurrent attention network for response generation. *arXiv preprint arXiv:1701.07149*.

Can Xu, Wei Wu, Chongyang Tao, Huang Hu, Matt Schuerman, and Ying Wang. 2019. Neural response generation with meta-words. *arXiv preprint arXiv:1906.06050*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Hainan Zhang, Yanyan Lan, Liang Pang, Jiafeng Guo, and Xueqi Cheng. 2019a. Recosa: Detecting the relevant contexts with self-attention for multi-turn dialogue generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3721–3730.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018a. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1108–1117.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018b. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2019b. Dialogpt: Large-scale generative pre-training for conversational response generation. *arXiv preprint arXiv:1911.00536*.

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.

Xueliang Zhao, Wei Wu, Chongyang Tao, Can Xu, Dongyan Zhao, and Rui Yan. 2020. Low-resource knowledge-grounded dialogue generation. *arXiv preprint arXiv:2002.10348*.

Yinhe Zheng, Rongsheng Zhang, Xiaoxi Mao, and Minlie Huang. 2019. A pre-training based personalized dialogue generation model with persona-sparse data. *arXiv preprint arXiv:1911.04700*.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2017. Emotional chatting machine: Emotional conversation generation with internal and external memory. *arXiv preprint arXiv:1704.01074*.