ProphetChat: Enhancing Dialogue Generation with Simulation of Future Conversation

Chang Liu[1,2], **Xu Tan**[3], **Chongyang Tao**[4], **Zhenxin Fu**[1], **Dongyan Zhao**[1,2,5,6]*,
Tie-Yan Liu[3], **Rui Yan**[7]*
[1]Wangxuan Institute of Computer Technology, Peking University
[2]Center for Data Science, Peking University
[3]Microsoft Research Asia [4]Microsoft Corporation
[5]Artificial Intelligence Institute of Peking University
[6]State Key Laboratory of Media Convergence Production Technology and Systems
[7]Gaoling School of Artificial Intelligence, Renmin University of China
{liuchang97,fuzhenxin,zhaody}@pku.edu.cn,
{xuta,chotao,tyliu}@microsoft.com ruiyan@ruc.edu.cn

Abstract

Typical generative dialogue models utilize the dialogue history to generate the response. However, since one dialogue utterance can often be appropriately answered by multiple distinct responses, generating a desired response solely based on the historical information is not easy. Intuitively, if the chatbot can foresee in advance what the user would talk about (i.e., the dialogue future) after receiving its response, it could possibly provide a more informative response. Accordingly, we propose a novel dialogue generation framework named ProphetChat that utilizes the simulated dialogue futures in the inference phase to enhance response generation. To enable the chatbot to foresee the dialogue future, we design a beam-search-like roll-out strategy for dialogue future simulation using a typical dialogue generation model and a dialogue selector. With the simulated futures, we then utilize the ensemble of a history-to-response generator and a future-to-response generator to jointly generate a more informative response. Experiments on two popular open-domain dialogue datasets demonstrate that ProphetChat can generate better responses over strong baselines, which validates the advantages of incorporating the simulated dialogue futures.

1 Introduction

Recent years have witnessed a surge of interest in building open-domain chatbots using generative approaches (Shang et al., 2015; Zhao et al., 2017; Tao et al., 2018; Zhang et al., 2020). These prevailing methods typically utilize dialogue histories as the dialogue context to generate the response via maximum likelihood estimation. Different from directed text generation tasks like machine translation where the target sentences are strictly constrained by the source sentence (Holtzman et al., 2019), the dialogue history and the response in chit-chat conversations are loosely coupled (Feng et al., 2020a). In other words, open-domain chatbots often have more “freedom" to decide what to respond since there often exists multiple distinct responses that can appropriately answer the given utterance. However, we argue that such excessive “freedom" also reveals that the dialogue history only may not contain enough information to generate a desired response that is informative and easy to reply to. If provided with enriched dialogue contexts that contain more useful dialogue cues, it could be easier for the model to chat with a human. So here comes the questions: what kind of dialogue cues is complementary to dialogue histories, and how to obtain and use them.

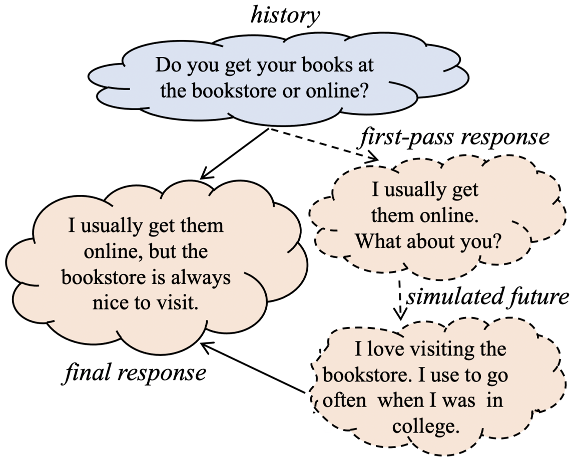
history
Do you get your books at the bookstore or online?
first-pass response
I usually get them online. What about you?
simulated future
I love visiting the bookstore. I use to go often when I was in college.
I usually get them online, but the bookstore is always nice to visit.
final response

Figure 1: A generation case of ProphetChat. It first simulates the dialogue future then generates the response conditioned on both the history and the future.

[*] Corresponding authors: Dongyan Zhao and Rui Yan.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 962 - 973

Recent studies in representation learning have demonstrated that when representing a token in a sentence, considering the tokens on its right side in addition to its left side can bring significant improvement (Devlin et al., 2019; Yang et al., 2019). Similar findings also appear in directed text generation where the future tokens on the right side can be beneficial to generate the current token (Serdyuk et al., 2017; Zhang et al., 2018c; Chen et al., 2020; Qi et al., 2020). Sharing the same spirit, we pursue to use the “right side" information, which is the dialogue future in our task, as the complementary dialogue cue to enhance the generation of the current response. Intuitively, if the chatbot can be told in advance what the user would probably talk about (i.e., the dialogue future) after receiving its response, it only needs to provide a response that can smoothly connect the history and the future. To verify whether the dialogue future can act as the complementary dialogue cue, we conduct empirical studies. We find that using a dialogue generation model to learn the reverse dialogue flow (i.e., using the future to generate the response) is quite effective. Furthermore, when utilizing the ensemble of the history-to-response generation model and the future-to-response generation model to generate the response conditioned on both the history and the gold future, the quality of the generated response surely improves. Though effective, the ground truth dialogue future is inaccessible in the inference phase. Therefore, all existing works in this line choose to leverage the dialogue future only in the training phase (Shen et al., 2018; Feng et al., 2020a,b), leaving the inference phase unchanged.

We argue that explicitly providing the possible dialogue futures in the inference phase can offer more direct help for the generation of the current response. To enable the incorporation of dialogue futures into response generation in the inference phase, we propose a response generation framework namely ProphetChat by answering two questions: how to acquire the future and how to use it. Figure 1 shows the generation process of ProphetChat. It consists of a history-to-response model (denoted as the forward model), a future-to-response model (denoted as the backward model), an ensemble gate, and a dialogue selector. Given a dialogue history, we first utilize an effective beam-search-like roll-out strategy to simulate possible dialogue futures. Concretely, the forward model first generates a batch of n possible responses based on the dialogue history. The dialogue selection model then comes to pick up the k-best responses. We further generate n possible futures for each of the picked responses, resulting in $k \cdot n$ futures. The selection model again picks up the k-best futures which are of higher quality compared with randomly sampled ones. Next, conditioned on both the history and the simulated future, we employ the forward model and the backward model to jointly generate the response by summing the per-step output probability distributions of the two models using a calculated weight. The weight is obtained by a trainable gate that learns to balance the trade-off between history and future information. Finally, we gather the k responses generated solely based on the history and the $k \cdot n$ responses generated based on both the history and the future, and use the selector to choose the top-ranked one as the final response. Since the ensemble generation model relies on the selector to sequentially select the response and the future, and the ensemble generation model also needs to learn how to balance the history and the future information given the selected future, we jointly train the whole model to make each module better collaborate with others to fulfill the ultimate goal: to maximize the likelihood of the gold response estimated by the ensemble generation model given the history and the selected future. We train the ensemble generation model directly using MLE objective while adopting reinforcement learning to tune the selector.

Our contributions in this paper are three folds:

- We propose a novel dialogue generation framework named ProphetChat which leverages the simulated dialogue future to enhance response generation through the ensemble of the history-to-response generator and the future-to-response generator. To the best of our knowledge, we are the first to utilize the dialogue futures for response generation in the inference phase.

- To acquire better dialogue futures in the inference phase, we propose an effective beam-search-like roll-out strategy for dialogue future simulation with the help of a dialogue selector.

- We conduct comprehensive experiments on two popular open-domain dialogue datasets and the results verify the advantages of incorporating the simulated dialogue futures.

2 Related Work

Dialogue System. Open-domain response generation has long been the research hot spot. Recently, various efforts have been made to generate informative and diverse responses by introducing effective architectures and learning objectives and by incorporating external knowledge. Zhao et al. (2017); Gu et al. (2018) applied CVAE to model the variability of responses. Li et al. (2016b); Zhang et al. (2018a); Saleh et al. (2020) adopted reinforcement learning to encourage the model to generate desired responses through carefully designed reward functions. Zhang et al. (2018b); Chan et al. (2019); Zheng et al. (2020); Li et al. (2021) exploited persona information to improve the coherence of the response. Zhou et al. (2018); Song et al. (2019); Shen and Feng (2020) considered emotions when generating the response. Dinan et al. (2018); Lian et al. (2019); Zhao et al. (2020a,b); Li et al. (2020) conditioned the response generation model with knowledge. Different from the above works that aimed to design specific history-to-response generation models, we propose a response generation framework where possible dialogue futures are utilized in the inference phase with the help of an effective future simulation strategy.

Future Modeling. There are various scenarios where considering future information is useful. In text generation, Serdyuk et al. (2017) proposed a twin network to regularize the hidden states of the left-to-right decoder with the future-aware right-to-left decoder. Zhang et al. (2018c) used the target-side hidden states generated by the right-to-left decoder to help the right-to-left decoder during translation so that the target-side future information can help avoid under-translation. Different from these works that consider the right side tokens as the future for the current token, we define “future" as the next dialogue utterance of the current response in a dialogue session. In response generation, Feng et al. (2020a) proposed to use gold futures as the conditions of two discriminators and adopted adversarial training to encourage diversity. Feng et al. (2020b) employed gold dialogue futures to learn a future-aware teacher model and transferred the knowledge to a history-to-response student model via imitation learning. These works only use the future information in the training phase, while we utilize the simulated dialogue future in the inference phase to provide the history-to-response generation model with direct help.

3 Preliminaries

In this section, we introduce the major off-the-shelf components in our framework.

DialoGPT (Zhang et al., 2020) is a GPT-based response generation model pre-trained on large-scale open-domain dialogue corpus by maximizing the likelihood of the successive dialogue utterances (i.e., the forward dialogue flow) given the initial dialogue history. While trained on the same corpus with the same architecture, DialoGPT-MMI is trained on the backward dialogue flow where the order of the utterances in a dialogue are reversed. We adopt DialoGPT as the forward generator and DialoGPT-MMI as the backward generator.

GRADE (Huang et al., 2020) is a graph-enhanced dialogue evaluation model that uses both utterance-level contextualized representations and topic-level graph representations to evaluate the response. As it is one of the SOTA dialogue evaluation models, we choose it as our dialogue selector.

4 Method

4.1 Overview

Our framework consists of a forward generator G_F that models the history-response-future dialogue flow, a backward generator G_B that models the reversed dialogue flow, a dialogue selector S that ranks the sampled utterances conditioned on the dialogue context, and a gate g that dynamically balance the ensemble weights between G_F and G_B. Given a history h, we first use G_F to sequentially sample the response r and the future f with the help of S. We then employ the ensemble of G_F and G_B using g to generate the response based on both h and f. The firstly generated responses together with the future-aware second-pass responses are finally re-ranked by S to produce the final response. Figure 2 illustrates our proposed framework.

4.2 Future Simulation

Given a dialogue history h, we first use G_F to generate n responses $\{r^i\}_{i=1}^n$ using top-k sampling (Fan et al., 2018). We denote these responses as the *first-pass responses*. Then the selector S calculates the quality scores $\boldsymbol{s_r} \in \mathbb{R}^n$ for all history-response pairs. The quality scores naturally form a propability distribution $\boldsymbol{p_r} \in \mathbb{R}^n$ over the sampled responses by using a softmax operation. We here consider the response selection procedure as sampling from such a distribution. Considering

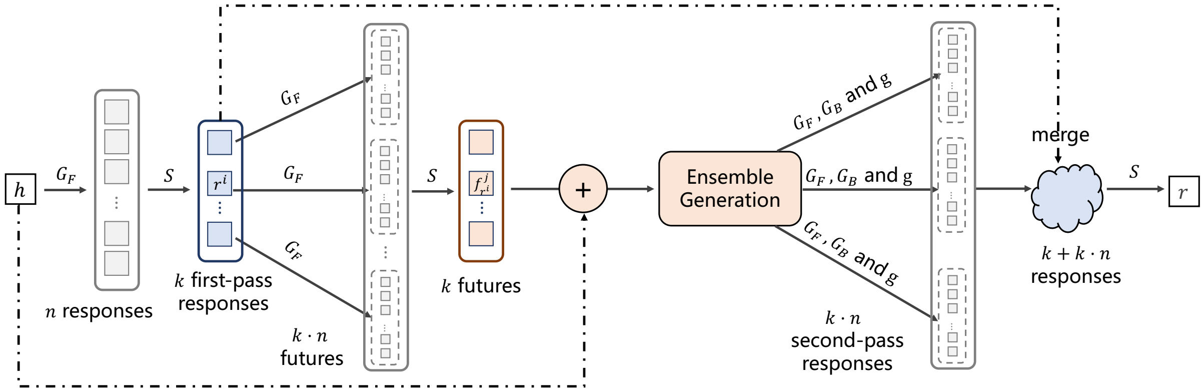

Figure 2: The overall framework of ProphetChat.

that the responses in open-domain dialogue are often diverse and hard to evaluate by any automatic evaluation metric, only using the response with the highest quality score or probability to further simulate the future is suboptimal. Meanwhile, generating futures conditioned on all the n sampled responses is too time-consuming. Therefore, borrowing the idea from beam search (Sutskever et al., 2014) where the k-best sentence prefixes are maintained during decoding to balance the searching performance and speed, we propose to keep the k-best responses at hand while discarding the others. For each of the selected response r^i, we concatenate it with h and use G_F to again sample n dialogue futures $\{f^j_{r^i}\}^n_{j=1}$, where $f^j_{r^i}$ denotes the j-th future simulated from h and r^i.

Up to now, we obtain $k \cdot n$ history-response-future dialogue triplets for the same dialogue history h by simulation. We again resort to the selector S to calculate the quality scores of all the generated futures conditioned on h and their corresponding ancestral responses as $\{\boldsymbol{s}_{\boldsymbol{f}_{\boldsymbol{r}^1}}, \ldots, \boldsymbol{s}_{\boldsymbol{f}_{\boldsymbol{r}^i}}, \ldots, \boldsymbol{s}_{\boldsymbol{f}_{\boldsymbol{r}^k}}\}$. We consider all the generated futures in the same sampling space (i.e., the future space of the given history) and directly perform softmax over the $k \cdot n$ quality scores to get the future distribution. Considering that the responses used to generate the futures are not equal in quality, we additionally multiply each probability of the simulated future $f^j_{r^i}$ with the probability of its ancestral response p_{r^i} to get the final ranking scores based on which we select k-best dialogue futures.

4.3 Ensemble Generation

Now with the history h and k plausible dialogue futures at hand, we pursue to generate the second-pass response conditioned on both the history and the future information. Given that the simulated futures contain noise derived from error accumulation in the simulation phase, it is necessary to balance the weights between the history-conditioned G_F and the future-conditioned G_B when they collaboratively generate the response. Hereby we introduce a trainable gate g which takes the last hidden states from G_F and G_B as inputs and calculates an ensemble weighting score w using an MLP with sigmoid activation. We then generate the response $\hat{r}$ using the per-step weighted ensemble of G_F and G_B conditioned on h and f:

$$\begin{aligned} P(\hat{r}_t|h, f, \hat{r}_{<t}; \theta_F, \theta_B, \theta_g) &= w \cdot P(\hat{r}_t|h, \hat{r}_{<t}; \theta_F) \\ &+ (1-w) \cdot P(\hat{r}_t|f, \hat{r}_{<t}; \theta_B), \end{aligned} \tag{1}$$

where the subscript t denotes the t-th token in $\hat{r}$ and θ_F, θ_B and θ_g denote the parameters of G_F, G_B and g respectively. Specifically, we sample n responses for the ensemble generation of h and each of the k futures, resulting in $k \cdot n$ future-aware responses. We denote these responses as the *second-pass responses*. To make full use of the k-best first-pass responses, we finally re-rank the $k + k \cdot n$ responses with S and consider the top-ranked response as our system outputs.

4.4 Training

Recall that there are several components G_F, G_B, S, and g in our framework. Although some of them can directly be used without post-training, this might be suboptimal. For one thing, post-training the models on domain-specific data with the same objective often brings better performance (Gururangan et al., 2020). For another, the original loss functions may not be thoroughly in accord with the ultimate goal in our framework. Thereby we propose a customized joint training algorithm.

For G_F and G_B, we adopt a similar training objective used by Zhang et al. (2020). Take G_F for example, we consider every consecutive three utterances in a dialogue session as a history-response-future triplet and fine-tune the models by minimizing the negative log likelihood of the response and the future conditioned on the history. G_B is fine-tuned in a similar manner with the reversed inputs. After fine-tuning, G_F and G_B are fixed.

For g, we can directly minimize the negative log-likelihood of the gold response r^*:

$$\mathcal{L}_1(\theta_g) = -\sum_t \log P(r_t^*|h, f, r_{<t}^*; \theta_F, \theta_B, \theta_g), \tag{2}$$

where f is simulated from either the gold response (denoted as the teacher-forcing mode) or a sampled response (denoted as the free-running mode).

While for S, considering that the original objective used in Huang et al. (2020) is not customized for selecting better responses and futures, it is better to perform task-specific post-training. Therefore, we propose to directly optimize S to our ultimate goal which is to maximize the log-likelihood of the gold response given the history and the selected simulated future. Since the sampling operation is non-differentiable, we use REINFORCE (Williams, 1992) with a self-critic (Rennie et al., 2017) baseline to estimate the gradient. We consider the future simulation process as sequential sampling from the score distributions of the responses and the futures respectively. Given the n responses generated by G_F conditioned on h, we sample a response r^i from $\boldsymbol{p}_r$. Then we generate n futures conditioned on h and r^i using G_F and again sample a future $f_{r^i}^j$ from them. We feed this sampled future and the gold history into our ensemble generation model and calculate the log-likelihood of the gold response, which is the opposite number of Equation 2, as the reward R. To reduce the variance of gradient estimation, we introduce a self-critic baseline. Concretely, we sequentially select the response and the future with the highest scores in each sampling step and calculate the reward of using the greedy future as the baseline reward R_b. The gradients are then estimated as follows:

$$\begin{aligned}\nabla_{\theta_S}\mathcal{L}_2(\theta_S) \approx -(R - R_b)\nabla_{\theta_S}[\log P(r^i|h;\theta_S) \\ + \log P(f_{r^i}^j|r^i, h; \theta_S)].\end{aligned} \tag{3}$$

Intuitively, directly forcing the model to learn in a fully free-running mode may be burdensome as the futures generated from the sampled responses may contain much noise. A better choice is to allow the model to gradually learn from easy to hard. We create a curriculum schedule (Bengio et al., 2015) that gradually switches from the teacher-forcing mode to the free-running mode. Specifically, let η denote the proportion of teacher-forcing mode, we gradually decrease η from β to α with cosine annealing schedule, where $0 \leq \alpha < \beta \leq 1$.

For the overall training, we first train g using Equation 2 and set $\eta = \beta$. Then we tune S using Equation 3 with the help of the above curriculum learning schedule. Finally, we jointly tune g and S with a fixed $\eta = \alpha$.

5 Experimental Setup

5.1 Datasets

To verify the effectiveness of our proposed response generation framework, we experiment on two popular dialogue datasets, **DailyDialog** (Li et al., 2017) and **PersonaChat** (Zhang et al., 2018b). We follow the original train/dev/test division and reconstruct the datasets by treating each consecutive three utterances as a triplet that represents history-response-future, resulting in approximately 65k/6k/6k examples in DailyDialog and 114k/14k/13k examples in PersonaChat.

5.2 Comparison Methods

5.2.1 Baselines

Posterior-GAN (Feng et al., 2020a) and **RegDG** (Feng et al., 2020b) are two non-GPT-based response generation models that use dialogue futures in the training time through either adversarial training or knowledge distillation.

DialoGPT$_F$ denotes the fine-tuned DialoGPT $_{medium}$ (Zhang et al., 2020) on two downstream datasets. **DialoGPT**$_{F,rerank}$ is its enhanced version which is equipped with the dialogue evaluation model (i.e., GRADE (Huang et al., 2020)) to select the top-ranked response.

5.2.2 Variants of ProphetChat

ProphetChat$_{k=?}$ denotes the model with the same model parameters but different beam sizes when simulating the futures.

ProphetChat$_{first}$ and **ProphetChat**$_{second}$ are used to denote the settings where only the first-pass or the second-pass responses are used in the final re-ranking process.

Models	B-1	B-2	B-3	B-4	D-1	D-2	D-3	D-4	AVG	EXT	GRE
Posterior-GAN	37.65	14.25	4.90	1.66	0.91	5.13	13.21	22.23	0.530	0.472	0.313
RegDG	38.77	14.36	5.13	1.91	1.07	5.95	14.85	24.78	0.550	**0.493**	0.319
DialoGPT$_{F}$	34.63	12.89	4.81	1.75	5.19	29.00	55.09	73.17	0.623	0.468	0.370
DialoGPT$_{F,rerank}$	34.66	12.99	4.87	1.77	6.79	36.59	64.75	81.18	0.612	0.456	0.369
ProphetChat	**39.33**	**14.57**	**5.38**	**1.93**	6.53	35.93	64.18	80.66	**0.626**	0.470	**0.372**
ProphetChat$_{first}$	37.58	14.00	5.22	1.89	6.49	35.44	63.18	79.73	0.625	0.468	0.369
ProphetChat$_{second}$	39.10	14.55	5.41	1.96	6.47	36.15	65.14	**81.92**	0.616	0.465	0.366
ProphetChat$_{k=1}$	35.27	13.17	4.92	1.79	**6.80**	**37.07**	**65.25**	81.39	0.612	0.464	0.368
ProphetChat$_{k=2}$	36.43	13.57	5.06	1.84	6.70	36.80	65.01	81.45	0.618	0.466	0.370
ProphetChat$_{k=3}$	37.71	14.04	5.22	1.89	6.59	36.27	64.68	81.24	0.622	0.468	0.370
ProphetChat *w/o* history	32.45	11.51	4.03	1.39	5.27	30.07	56.94	74.35	0.601	0.442	0.347
ProphetChat *w/o* selector	35.74	13.08	4.75	1.68	5.07	28.53	54.98	73.12	0.623	0.464	0.364
ProphetChat *w/o* train	38.87	14.06	5.20	1.88	6.31	35.33	63.01	79.52	0.623	0.465	0.367
ProphetChat *w* gold future	39.00	14.43	5.34	1.93	4.91	28.80	56.22	74.57	0.640	0.477	0.376

Models	B-1	B-2	B-3	B-4	D-1	D-2	D-3	D-4	AVG	EXT	GRE
Posterior-GAN	44.13	16.57	5.73	1.91	0.41	2.18	5.34	9.91	0.647	0.489	0.380
RegDG	46.12	17.11	5.90	2.01	0.43	2.41	6.26	11.55	0.653	**0.512**	0.381
DialoGPT$_{F}$	45.84	16.91	6.07	2.12	2.26	14.89	32.28	48.35	0.657	0.480	0.383
DialoGPT$_{F,rerank}$	46.69	17.18	6.13	2.13	2.85	19.40	41.96	61.69	0.657	0.481	0.386
ProphetChat	47.55	**17.50**	**6.26**	**2.19**	3.01	20.01	42.32	61.58	**0.662**	0.484	**0.393**
ProphetChat$_{first}$	47.51	17.47	6.23	2.17	2.86	19.15	40.99	60.46	0.660	0.483	0.390
ProphetChat$_{second}$	46.43	17.03	6.05	2.10	**3.06**	**20.81**	**43.90**	**63.67**	0.661	0.484	0.391
ProphetChat$_{k=1}$	46.44	17.08	6.10	2.12	3.01	20.34	43.36	63.12	0.659	0.482	0.390
ProphetChat$_{k=2}$	46.92	17.20	6.15	2.14	3.05	20.18	42.91	62.56	0.659	0.483	0.391
ProphetChat$_{k=5}$	**47.66**	17.49	6.12	2.15	3.00	19.87	41.95	61.14	0.658	0.482	0.388
ProphetChat *w/o* history	42.47	15.23	5.30	1.81	2.42	15.84	34.74	52.70	0.637	0.461	0.369
ProphetChat *w/o* selector	46.38	16.98	6.05	2.11	2.30	15.44	34.35	52.32	0.656	0.477	0.382
ProphetChat *w/o* train	47.44	17.23	6.16	2.14	2.91	19.58	41.69	60.80	0.659	0.480	0.390
ProphetChat *w* gold future	48.28	17.99	6.57	2.34	2.36	15.87	35.23	53.37	0.668	0.492	0.393

Table 1: Response generation results on DailyDialog (the upper) and PersonaChat (the lower) datasets. Within each table, the upper block lists baseline results, the middle block presents the performance of ProphetChat and its variants, and the lower block gives the ablation results. The lines with gray backgound are our main model.

5.2.3 Ablations of ProphetChat

ProphetChat *w/o* history means we utilize the top-ranked simulated future to generate the response without the help of the history.

ProphetChat *w/o* selector denotes the model where we sequentially sample the responses and the futures randomly without using the selector.

ProphetChat *w/o* train means we directly utilize the fine-tuned G_F, G_B and the fixed S without post-training. We manually choose a fixed ensemble weight for the ensemble generation process instead of using a trainable gate.

ProphetChat *w/* gold future denotes the model that utilizes the history and the gold future, which is **inaccessible** in the inference phase, to generate the response.

5.3 Implementation Details

Our implementation is based on the open-source toolkit Transformers (Wolf et al., 2020). For the generator G_F and G_B, we initialize them with the publicly released DialoGPT$_{medium}$ and DialoGPT-MMI$_{medium}$[1]. For the dialogue selector, we use the pre-trained GRADE[2] as initialization. We firstly use AdamW (Loshchilov and Hutter, 2017) with learning rate 3e-5 to fine-tune G_F and G_B. Then, we jointly train the ensemble gate g and the top non-transformer layers of the selector S with learning rate 2e-5, while keeping other parameters (i.e., G_F, G_B and most of the parameters of S except for its top layers) fixed. We set the curriculum hyperparameters (α, β) as (0.0, 1.0) on both datasets. We fix the sample number n of both the response and the future as 10 and vary the simulation beam size $k \in \{1, 2, 3, 5\}$. We choose k=5 on DailyDialog and k=3 on PersonaChat. We use top-k sampling (Fan et al., 2018) to generate the first-pass responses, the futures, and the second-pass responses with the temperature as 0.7 and k as 40. All the hyperparameters are chosen depending on

[1] https://github.com/microsoft/DialoGPT

[2] https://github.com/li3cmz/GRADE

Models	Readability	kappa	Sensibleness	kappa	Specificity	kappa
Posterior-GAN	0.58	0.42	0.46	0.49	0.21	0.58
RegDG	0.60	0.45	0.51	0.59	0.27	0.50
DialoGPT$_F$	0.68	0.52	0.64	0.61	0.44	0.52
DialoGPT$_{F,rerank}$	0.69	0.50	0.69	0.60	0.45	0.64
ProphetChat	**0.71**	0.52	**0.75**	0.53	**0.49**	0.49
Models	Readability	kappa	Sensibleness	kappa	Specificity	kappa
Posterior-GAN	0.64	0.58	0.50	0.65	0.24	0.48
RegDG	0.65	0.56	0.53	0.55	0.28	0.63
DialoGPT$_F$	0.69	0.59	0.68	0.66	0.42	0.52
DialoGPT$_{F,rerank}$	0.70	0.44	0.72	0.52	0.48	0.52
ProphetChat	**0.72**	0.43	**0.77**	0.61	**0.53**	0.54

Table 2: Human evaluation results on DailyDialog (the upper) and PersonaChat (the lower) datasets.

Models	B-1	B-2	B-3	B-4	D-1	D-2	D-3	D-4	AVG	EXT	GRE
TF$_{rerank}$	**39.94**	**14.89**	**5.51**	**1.98**	**6.63**	**37.95**	**68.03**	**84.52**	**0.630**	**0.475**	**0.380**
FR$_{k=1}$	30.19	10.74	3.76	1.29	5.61	31.64	59.16	77.30	0.589	0.431	0.337
FR$_{k=2}$	38.10	13.52	4.72	1.61	6.50	37.12	66.98	83.88	0.588	0.430	0.332
FR$_{k=3}$	38.70	13.73	4.79	1.64	6.55	37.26	67.14	84.00	0.587	0.429	0.331
FR$_{k=5}$	39.13	13.87	4.83	1.65	6.54	37.28	67.13	83.91	0.587	0.429	0.330
Models	B-1	B-2	B-3	B-4	D-1	D-2	D-3	D-4	AVG	EXT	GRE
TF$_{rerank}$	42.89	15.51	**5.46**	**1.87**	**3.37**	**23.62**	**50.15**	**71.23**	0.616	0.446	**0.365**
FR$_{k=1}$	43.59	15.46	5.31	1.80	2.36	16.32	36.51	55.72	**0.630**	**0.450**	0.354
FR$_{k=2}$	44.71	15.75	5.37	1.81	2.88	20.77	44.85	65.21	0.629	0.447	0.354
FR$_{k=3}$	44.70	15.74	5.36	1.80	2.89	20.92	45.08	65.48	0.628	0.446	0.352
FR$_{k=5}$	**44.74**	**15.76**	5.37	1.81	2.91	20.99	45.13	65.49	0.627	0.446	0.352

Table 3: Future simulation results on DailyDialog (the upper) and PersonaChat (the lower) datasets. TF$_{rerank}$ (i.e., teacher forcing) means using the history and the gold response to generate the future then re-ranking using the selector. FR$_{k=?}$ (i.e., free running) means using our proposed future simulation methods.

their performance on the development set.

5.4 Evaluation Metrics

Automatic Metrics. We use **BLEU** (Papineni et al., 2002) to measure the word overlap between the ground truth responses and the generated ones. For simplification, we use B-n to denote the n-gram overlap scores. We employ **Distinct 1-4** (Li et al., 2016a) to measure the diversity of the generated responses, where Distinct-n (abbreviated as D-n) represents the ratio of distinct n-grams in responses. We adopt the embedding-based metrics (i.e., **Average**, **Extrema**, and **Greedy**) (Liu et al., 2016) to measure the semantic relevance between the ground truth responses and the generated ones.
Human Evaluation. We ask three well-educated annotators to score 150 randomly selected responses generated by ProphetChat and other baselines. The annotators are asked to evaluate the human-likeness of the responses from three perspectives: readability, sensibleness and specificity. For readability, we ask annotators whether the response is grammatically correct and easy to read. For sensibleness and specificity, we follow Adiwardana et al. (2020) to conduct the evaluation. For all three metrics, the annotators are asked to give 0-1 labels. We provide the averaged scores and further calculate the Fleiss's kappa (Fleiss, 1971) to measure the inter-annotator agreement.

6 Experimental Results

6.1 Overall Performance

Table 1 presents the overall performance of our proposed method as well as its variants and ablations. Table 2 shows the human evaluation results. Compared with the two non-GPT baselines, GPT-based models generally achieve superior performance, especially in Distinct and human evaluation. ProphetChat outperforms all the baseline methods by a large margin on both datasets in almost all automatic metrics and all human evaluation metrics. For human evaluation results, the Fleiss's kappa scores are mainly distributed in [0.4, 0.6], which means annotators achieved moderate agreement.

6.2 Discussion of Model Variants

With the same model parameters, we have several model variants by using different hyperparameters or computation flow in the inference phase. Here we mainly discuss two types of model variants: (1) the model with different simulation beam size k, (2) the final re-ranking among the first-pass responses or the second-pass responses.

The simulation beam size. When simulating the dialogue future, we can choose different beam sizes to balance the computation cost and the performance. We test $k \in \{1, 2, 3, 5\}$ on both datasets and find $k = 5$ is better than others on DailyDialog, while $k = 3$ is enough on PersonaChat. It can be seen that when k is small, increasing k can boost the performance. With the appropriate choice of the simulation beam size, ProphetChat can be deployed to various scenarios with different computation resources. We further directly test the future simulation performance by comparing the futures generated by our method and generated using the gold responses. The results are listed in Table 3. It can be observed that on DailyDialog, with the increase of k, our future simulation method gradually catches up with the teacher forcing counterpart in BLEU and Distinct, while still lagging behind in embedding-based metrics. On PersonaChat, our method even outperforms TF$_{rerank}$ in several metrics. These findings proves that we are able to obtain dialogue futures of good quality solely based on the history through our effectiveness future simulation algorithm.

Re-ranking among the first pass or the second pass responses. Recall that in our main framework, we finally gather the k first-pass responses and the $k \cdot n$ second-pass responses together and finally re-rank them with the selector conditioned on both the history and the corresponding future. From Table 1 we can find that on both datasets, re-ranking using both groups of responses yield better performance in most of the metrics than only using one of them. When comparing their individual performance, it can be observed that on DailyDialog, ProphetChat$_{second}$ is superior to ProphetChat$_{first}$ in BLEU and Distinct, while ProphetChat$_{first}$ wins in embedding-based metrics. On PersonaChat, ProphetChat$_{second}$ wins all metrics except BLEU. There exist some cases where the simulated futures are meaningless or include irrelevant information. When this happens, the final re-ranking process comes as the remedy. We find that the proportions of the test cases where the final responses are picked from the second-pass responses are 40.4% on DailyDialog and 36.6% on PersonaChat, which are less than the proportions of the second-pass responses involved in re-ranking. This finding indicates that re-ranking plays a vital role to select the appropriate responses from the two groups of candidate responses of various qualities.

6.3 Ablation Study

We make ablation study from several perspectives including the effect of the history, the selector and the training algorithm. Table 1 shows that although only using the simulated future (i.e., ProphetChat *w/o* history) can generate plausible responses, the performance is largely inferior to the full model. Also, we observe that ProphetChat *w/o* selector underperforms the full model, demonstrating the effectiveness of the selector which helps simulate better futures. When considering the training objective, we find that ProphetChat *w/o* train already achieves good performance, but jointly training the whole model further makes our model perform better. Finally, when provided with the gold future, ProphetChat *w/* gold future outperforms ProphetChat in terms of embedding-based metrics on both datasets, and BLEU on PersonaChat, while underperforming on other metrics. In other words, with the simulated futures, ProphetChat can achieve comparable performance with the model that "cheats" to see the gold future, which also demonstrates the effectiveness of our method.

6.4 Case Study

Figure 3 presents two cases sampled from the two datasets. For ProphetChat, in addition to its final response, we provide its corresponding first-pass response, and the simulated future of the response. From the two cases we can observe that by taking the simulated future into consideration, ProphetChat generates more informative responses than baselines. Specifically, in case 1, when a two-choice query is issued in the history, the first-pass response chooses “online" as the answer and meanwhile poses another question. ProphetChat then uses the history and this first-pass response to simulate the future where the other possible choice (i.e., the bookstore) is talked about. Given the history and the simulated future, ProphetChat finally obtains its response which not only answers the query

Case 1

History:That is cool. Do you get your books at the bookstore or online?
Gold Response: I usually try to use the library, but otherwise I get them online.
Gold Future: Are you going for elementary education or high school or college?

PropherChat: I usually get them online, but the bookstore is always nice to visit.
PropherChat First-Pass Response: I usually get them online. What about you?
Simulated Future: I love visiting the bookstore. I use to go often when I was in college.

DialoGPT$_F$: I usually get them online. You?
Posterior-GAN: I get them on line.
RegDG: I like to go to the bookstore.

Case 2

History:Well, I'm a bit out of shape. I'm thinking about getting some exercises to keep fit.
Gold Response: Oh, that's good news for us.
Gold Future: So what do you provide?

PropherChat: That's a good idea! What kind of exercise do you like to do?
PropherChat First-Pass Response: What kind of exercise?
Simulated Future: Well, I'm thinking of doing some kind of body weight exercises. I don't know if that will help me lose weight, though.

DialoGPT$_F$: What kind of exercise?
Posterior-GAN: Yes, you need exercise.
RegDG: I'd like to do some exercise too.

Figure 3: Generation cases from the two datasets. Case 1 is sampled from PersonaChat and Case 2 is sampled from DailyDialog.

in the history but also incorporates the cues in the future. This response becomes more informative than the previous one. A similar phenomenon can also be found in case 2 where the final response is more comprehensive that connects the history and the future smoothly.

7 Conclusion

We propose a novel response generation framework that utilizes the simulated dialogue futures in the inference phase to enhance response generation. To acquire the dialogue futures, we design an effective beam-search-like roll-out strategy using a history-to-response dialogue generation model and a dialogue selector. To make use of the simulated future, we use the dynamic ensemble of the history-to-response and the future-to-response generation model. Experiment results demonstrate the effectiveness of our proposed method on two popular datasets. In the future, we plan to enable our future simulation method to simulate multiple turns of dialogue futures.

Acknowledgements

We would like to thank the anonymous reviewers for their constructive comments. This work was supported by the National Key Research and Development Program of China (No. 2020AAA0106600), National Natural Science Foundation of China (NSFC Grant No. 62122089 and No. 61876196), and Beijing Outstanding Young Scientist Program (No. BJJWZYJH012019100020098).

Ethical Statement

This paper proposes a new dialogue generation framework that utilizes the simulated dialogue futures in the inference phase to enhance the generation of the response. Generative approaches are widely used in a wide range of dialogue applications. The proposed method improves the quality of the generated responses, which could be beneficial to research and real-world applications. The research will not pose ethical issues. This paper doesn't involve any data collection and release thus there are no privacy issues. All the datasets used in this paper are publicly available and are widely adopted by researchers to test the performance of open-domain response generation models. This paper conducts human evaluation to evaluate the quality of the generated responses. Three part-time research assistants were recruited to do human evaluation with clearly demonstrated evaluation rules. They worked with the pay 100 CNY/hour during their evaluation.

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*.

S. Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. 2015. Scheduled sampling for sequence prediction with recurrent neural networks. In *NIPS*.

Zhangming Chan, Juntao Li, Xiaopeng Yang, Xiuying Chen, Wenpeng Hu, Dongyan Zhao, and Rui Yan. 2019. Modeling personalization in continuous space for response generation via augmented Wasserstein autoencoders. 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 1931–1940, Hong Kong, China. Association for Computational Linguistics.

Yen-Chun Chen, Zhe Gan, Yu Cheng, Jingzhou Liu, and Jingjing Liu. 2020. Distilling knowledge learned in BERT for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7893–7905, Online. Association for Computational Linguistics.

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.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 889–898, Melbourne, Australia. Association for Computational Linguistics.

Shaoxiong Feng, Hongshen Chen, Kan Li, and Dawei Yin. 2020a. Posterior-gan: Towards informative and coherent response generation with posterior generative adversarial network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 7708–7715.

Shaoxiong Feng, Xuancheng Ren, Hongshen Chen, Bin Sun, Kan Li, and Xu Sun. 2020b. Regularizing dialogue generation by imitating implicit scenarios. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6592–6604, Online. Association for Computational Linguistics.

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

Xiaodong Gu, Kyunghyun Cho, Jung-Woo Ha, and Sunghun Kim. 2018. Dialogwae: Multimodal response generation with conditional wasserstein autoencoder. *arXiv preprint arXiv:1805.12352*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

Lishan Huang, Zheng Ye, Jinghui Qin, Liang Lin, and Xiaodan Liang. 2020. GRADE: Automatic graph-enhanced coherence metric for evaluating open-domain dialogue systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9230–9240, Online. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. 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, San Diego, California. Association for Computational Linguistics.

Jiwei Li, Will Monroe, Alan Ritter, Dan Jurafsky, Michel Galley, and Jianfeng Gao. 2016b. Deep reinforcement learning for dialogue generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1192–1202, Austin, Texas. Association for Computational Linguistics.

Juntao Li, Chang Liu, Chongyang Tao, Zhangming Chan, Dongyan Zhao, Min Zhang, and Rui Yan. 2021. Dialogue history matters! personalized response selection in multi-turn retrieval-based chatbots. *ACM Transactions on Information Systems (TOIS)*, 39(4):1–25.

Linxiao Li, Can Xu, Wei Wu, Yufan Zhao, Xueliang Zhao, and Chongyang Tao. 2020. Zero-resource knowledge-grounded dialogue generation. *Advances in Neural Information Processing Systems*, 33:8475–8485.

Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 986–995, Taipei, Taiwan. Asian Federation of Natural Language Processing.

Rongzhong Lian, Min Xie, Fan Wang, Jinhua Peng, and Hua Wu. 2019. Learning to select knowledge for response generation in dialog systems. *arXiv preprint arXiv:1902.04911*.

Chia-Wei Liu, Ryan Lowe, Iulian Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2122–2132, Austin, Texas. Association for Computational Linguistics.

I. Loshchilov and F. Hutter. 2017. Fixing weight decay regularization in adam. *ArXiv*, abs/1711.05101.

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, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting future n-gram for sequence-to-SequencePre-training. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2401–2410, Online. Association for Computational Linguistics.

Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7008–7024.

Abdelrhman Saleh, Natasha Jaques, Asma Ghandeharioun, Judy Shen, and Rosalind Picard. 2020. Hierarchical reinforcement learning for open-domain dialog. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8741–8748.

Dmitriy Serdyuk, Nan Rosemary Ke, Alessandro Sordoni, Adam Trischler, Chris Pal, and Yoshua Bengio. 2017. Twin networks: Matching the future for sequence generation. *arXiv preprint arXiv:1708.06742*.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1577–1586, Beijing, China. Association for Computational Linguistics.

Lei Shen and Yang Feng. 2020. CDL: Curriculum dual learning for emotion-controllable response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 556–566, Online. Association for Computational Linguistics.

Xiaoyu Shen, Hui Su, Wenjie Li, and Dietrich Klakow. 2018. NEXUS network: Connecting the preceding and the following in dialogue generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4316–4327, Brussels, Belgium. Association for Computational Linguistics.

Zhenqiao Song, Xiaoqing Zheng, Lu Liu, Mu Xu, and Xuanjing Huang. 2019. Generating responses with a specific emotion in dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3695, Florence, Italy. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. *Advances in neural information processing systems*, 27.

Chongyang Tao, Shen Gao, Mingyue Shang, Wei Wu, Dongyan Zhao, and Rui Yan. 2018. Get the point of my utterance! learning towards effective responses with multi-head attention mechanism. In *IJCAI*, pages 4418–4424.

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Hainan Zhang, Yanyan Lan, Jiafeng Guo, Jun Xu, and Xueqi Cheng. 2018a. Reinforcing coherence for sequence to sequence model in dialogue generation. In *IJCAI*, pages 4567–4573.

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? In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2204–2213, Melbourne, Australia. Association for Computational Linguistics.

Xiangwen Zhang, Jinsong Su, Yue Qin, Yang Liu, Rongrong Ji, and Hongji Wang. 2018c. Asynchronous bidirectional decoding for neural machine translation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 270–278, Online. Association for Computational Linguistics.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 654–664, Vancouver, Canada. Association for Computational Linguistics.

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

Xueliang Zhao, Wei Wu, Can Xu, Chongyang Tao, Dongyan Zhao, and Rui Yan. 2020b. Knowledge-grounded dialogue generation with pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3377–3390, Online. Association for Computational Linguistics.

Yinhe Zheng, Rongsheng Zhang, Minlie Huang, and Xiaoxi Mao. 2020. A pre-training based personalized dialogue generation model with persona-sparse data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9693–9700.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.