Diverse and Faithful Knowledge-Grounded Dialogue Generation via Sequential Posterior Inference

Yan Xu [1 †] **Deqian Kong** [2 †] **Dehong Xu** [2] **Ziwei Ji** [1] **Bo Pang** [3] **Pascale Fung** [1 ‡] **Ying Nian Wu** [2 ‡]

Abstract

The capability to generate responses with diversity and faithfulness using factual knowledge is paramount for creating a human-like, trustworthy dialogue system. Common strategies either adopt a two-step paradigm, which optimizes knowledge selection and response generation separately, and may overlook the inherent correlation between these two tasks, or leverage conditional variational method to jointly optimize knowledge selection and response generation by employing an inference network. In this paper, we present an end-to-end learning framework, termed *Sequential Posterior Inference* (SPI), capable of selecting knowledge and generating dialogues by approximately sampling from the posterior distribution. Unlike other methods, SPI does not require the inference network or assume a simple geometry of the posterior distribution. This straightforward and intuitive inference procedure of SPI directly queries the response generation model, allowing for accurate knowledge selection and generation of faithful responses. In addition to modeling contributions, our experimental results on two common dialogue datasets (Wizard of Wikipedia and Holl-E) demonstrate that SPI outperforms previous strong baselines according to both automatic and human evaluation metrics.The code and checkpoints are available at `https://github.com/deqiankong/SPI`.

[†]Equal contribution [‡]Equal advising [1]Center for Artificial Intelligence Research (CAiRE), The Hong Kong University of Science and Technology, Hong Kong [2]Department of Statistics, UCLA, CA, USA [3]Salesforce Research. Correspondence to: Yan Xu <yxucb@connect.ust.hk>, Deqian Kong <deqiankong@ucla.edu>.

Proceedings of the 40th International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023. Copyright 2023 by the author(s).

1. Introduction

Open-domain dialogue systems aim at fulfilling human-machine conversations by producing human-like responses to utterances from humans (Serban et al., 2016). The emergence of large-scale pre-trained language models (PLMs) has turbocharged the development of open-domain dialogue systems (Zhang et al., 2020; Roller et al., 2021). By maximizing the token-level likelihood of gold responses given dialogue history, dialogue systems can generate fluent and natural responses. However, challenges remain to ensure that responses are diverse and informative (Ghazvininejad et al., 2018), yet remain factual and accurate (Shuster et al., 2021). Prior approaches for improving the diversity of dialogue responses focus on preventing them from being dull and repetitive (Zhao et al., 2019; Xu et al., 2022), while optimizing for diversity alone tends to encourage the dialogue system to hallucinate non-factual responses (Ji et al., 2022a). ChatGPT (OpenAI, 2023) tries to address this issue using a reward model trained with human preference. However, it is very resource-consuming. To address this limitation in generative dialogue systems, we need to ground system responses on external knowledge effectively.

Knowledge-grounded dialogue (KGD) has been investigated in recent years (Dinan et al., 2018; Li et al., 2020; Xu et al., 2021; Yang et al., 2022). The objective is to enhance dialogue response generation to facilitate engaging and in-depth conversations, while avoiding the inclusion of non-factual information. The task can be achieved following a two-step paradigm: (1) knowledge selection; (2) response generation. Some previous works (Lian et al.; Kim et al., 2019; Chen et al., 2020) optimize these two steps individually. They first utilize variational inference (Kingma & Welling, 2014) for knowledge selection, where the prior distribution is conditioned on dialogue history, and the posterior distribution depends on both response and dialogue history. Then they optimize the response generation task based on the selected knowledge. Since knowledge selection in KGD tasks is a complex one-to-many problem, it is not trivial to generate a factual response with dialogue history and selected knowledge solely, not to mention that inaccurate knowledge may be chosen even with a complex knowledge selection module. Other works (Liu et al., 2021) bypass the

knowledge selection step by providing all the knowledge candidates to the response generator, which is computationally inefficient. Therefore, it is natural to choose a probabilistic model with two latent variables to select knowledge and generate responses so that both procedures can be optimized simultaneously. CoLV (Zhan et al., 2021) follows this scheme and chooses to optimize these latent variables by recruiting an inference network to infer the posterior distribution. However, such methods using variational inference trained with evidence lower bound (ELBO) may ignore the fact that knowledge selection is inherently correlated to response generation. Hence, there might be a large amortization gap between log-likelihood and ELBO (Cremer et al., 2018). An alternative to variational inference is posterior inference, such as Markov Chain Monte Carlo (MCMC) which may be in the form of Langevin dynamics (Langevin, 1908). (Pang et al., 2021a) proposes to generate text using short-run inference dynamics, such as finite step Langevin dynamics guided by the posterior distribution of the latent variable. Posterior inference has demonstrated its simplicity and superiority in image modeling, trajectory prediction, etc. (Pang et al., 2020; 2021b; Xie et al., 2022; Li & Han, 2022). However, Posterior inference based methods are still under-explored in the scenarios of PLMs.

In this work, we propose a probabilistic model with dual latent variables, a discrete latent variable for knowledge selection, and a continuous latent variable for response generation. Instead of variational inference, we propose a new approximate sampling method, *Sequential Posterior Inference* (SPI). This model can be learned by approximate maximum likelihood estimation (MLE). Compared to variational inference, SPI has the advantage of fewer model parameters since there is no need to parameterize the inference network, which eases the effort of fine-tuning in PLMs. To amplify the efficiency of SPI within PLMs, we propose to leverage the initializer or learnable prior to sample the discrete latent variable, and short-run MCMC to sample the continuous latent variable. Empirically, we show that the model trained with SPI can generate faithful and diverse responses with external knowledge. Our model outperforms previous methods on both WoW and Holl-E benchmarks. Further human evaluation has demonstrated its superiority as well.

Our contributions are three-fold:

(1) We propose a probabilistic dialogue system for KGD that can be learned by approximate MLE with sequential posterior inference (SPI).

(2) We propose to use an initializer and short-run MCMC to explore the discrete and continuous search spaces, which enables efficient approximate MLE learning in PLMs.

(3) Our proposed model achieves state-of-the-art (SOTA) performance on two common KGD benchmarks.

2. Methods

2.1. Model

Suppose we have N observed examples $\{\mathcal{D}^n\}_{n=1}^N$ in dialogue dataset. For each example, $\mathcal{D}^n = (C^n, R^n)$, where C^n is the dialogue context, and R^n is the response based on the dialogue history and selected knowledge. In KGD tasks, each dialogue context consists of dialogue history H^n, and a set of M knowledge candidates $\mathbf{K}^n = \{K_i^n\}_{i=1}^M$, denoted as $C^n = (H^n, \mathbf{K}^n)$.

We consider the KGD task as a conditional generation process given dialogue history. Let $s \in \{1, \dots, M\}$ be a discrete variable indicating the choice of the knowledge candidate. Let $z \in \mathbb{R}^d$ be a d-dimensional continuous variable as a summary or abstraction of the future response to account for the sentence-level semantics. Consider the following generative model for R,

$$(s, z) \sim p_\alpha(s, z|C), \quad R \sim p_\beta(R|s, z, C), \tag{1}$$

where $p_\alpha(s, z|C)$ is the context-conditioned prior model parameterized by α and $p_\beta(R|s, z, C)$ is the response generation model parameterized by β.

To be specific, we may factorize the context-conditioned prior model as

$$s \sim p_{\alpha_1}(s|C), \quad z \sim p_{\alpha_2}(z|s, C), \tag{2}$$

where $p_{\alpha_1}(s|C)$ can be defined as a simple uniform distribution $\mathbb{P}(s = i) = \frac{1}{M}, i \in \{1, \dots, M\}$, or a learnable distribution $\mathbb{P}(s = i) = \frac{\exp(f_{\alpha_1}(s=i,C))}{\sum_{s=1}^M \exp(f_{\alpha_1}(s,C))}, i \in \{1, \dots, M\}$, and $p_{\alpha_2}(z|s, C) = \mathcal{N}(f_{\alpha_2}(s, C), \mathbf{I})$ is isotropic Gaussian. In our implementation, $f_\alpha(\cdot)$ is parameterized based on a pre-trained BART (Lewis et al., 2020) encoder and $\alpha = (\alpha_1, \alpha_2)$ consists of the parameters of two priors.

For the response generation model, $p_\beta(R|s, z, C)$ is defined in a conditional auto-regressive manner,

$$p_\beta(R|s, z, C) = \prod_{l=1}^{L} p_\beta(r_l|s, z, r_{<l}, C), \tag{3}$$

where L refers to the sentence length of the response R, r_l is the l-th token of the response, and p_β is parameterized based on a pre-trained BART decoder. Note that s and z control every step of the auto-regressive generation.

The context-conditioned distribution of response R is $p_\theta(R|C) = \sum_s \int p_\theta(s, z, R|C)dz$, where $\theta = \{\alpha, \beta\}$. Given C and R, the inference of (s, z) can be approximately achieved using SPI (see Section 2.3),

$$p_\theta(s, z|R, C) = \frac{p_\theta(s, z, R|C)}{p_\theta(R|C)}. \tag{4}$$

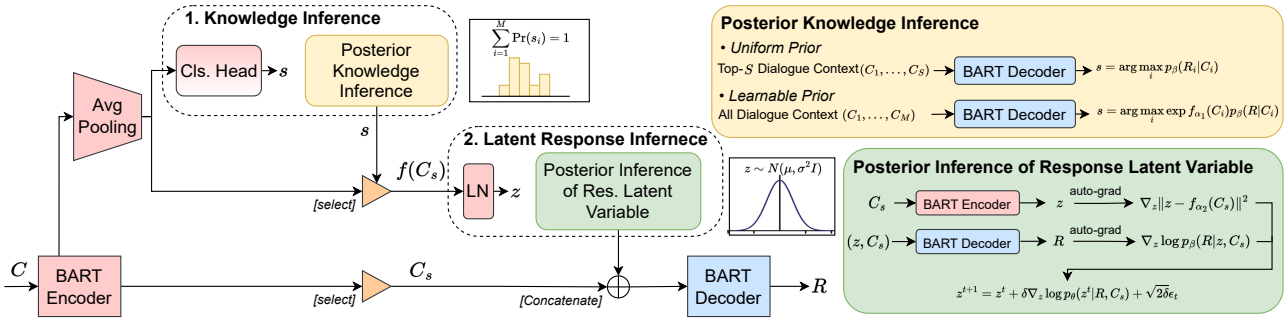

Figure 1. The overview of the learning algorithm of **SPI** (left), where modules in pink denote the context-conditioned prior model and modules in blue denote the response generation model. The prior model is mainly instantiated with the BART encoder, while the generator is implemented with the BART decoder. We also demonstrate details of posterior knowledge selection and posterior inference of the response latent variable on the right.

2.2. Learning

Given training examples, $\{\mathcal{D}^n = (C^n, R^n)\}_{n=1}^N$, the model can be learned using maximum likelihood estimation (MLE) where the log-likelihood is

$$L(\theta) = \frac{1}{N}\sum_{n=1}^{N} \log p_\theta(R^n|C^n), \tag{5}$$

where $\theta = \{\alpha, \beta\}$ are the learnable parameters of the model.

Then the gradient of the log-likelihood function can be calculated by

$$\begin{aligned}
&\nabla_\theta \log p_\theta(R|C) \\
&= \frac{1}{p_\theta(R|C)} \nabla_\theta p_\theta(R|C) \\
&= \frac{1}{p_\theta(R|C)} \sum_s \int \nabla_\theta p_\theta(s, z, R|C) dz \\
&= \sum_s \int \frac{p_\theta(s, z, R|C)}{p_\theta(R|C)} \nabla_\theta \log p_\theta(s, z, R|C) dz \\
&= \mathbb{E}_{p_\theta(s,z|R,C)}[\nabla_\theta \log p_\theta(s, z, R|C)].
\end{aligned} \tag{6}$$

Although the context-conditioned distribution $p_\theta(R|C)$ is intractable due to the latent variables being integrated out, we can approximate the above expectation using Monte Carlo samples from the posterior $p_\theta(s, z|R, C)$ in Equation (6), which will be further discussed in details as sequential posterior inference in Section 2.3.

For the gradient of the log-likelihood, we have

$$\begin{aligned}
&\nabla_\theta \log p_\theta(s, z, R|C) \\
&= \nabla_\theta \log p_\alpha(s, z|C) + \nabla_\theta \log p_\beta(R|s, z, C).
\end{aligned} \tag{7}$$

For discrete variable s, if we assume a simple uniform prior distribution, $\nabla_\theta \log p_\alpha(s, z|C) = \nabla_\theta \log p_{\alpha_2}(z|s, C)$. Otherwise, $\nabla_\theta \log p_\alpha(s, z|C)$ becomes $\nabla_\theta \log p_{\alpha_1}(s|C) + \nabla_\theta \log p_{\alpha_2}(z|s, C)$ if we assume a learnable prior distribution.

2.3. Sequential Posterior Inference

In Equation (6), the expectation can be approximated by Monte Carlo average over samples (s, z) from $p_\theta(s, z|R, C)$. We define the pair between dialogue history H and the s-th index of knowledge in $\mathbf{K}$ as $C_s = (H, K_s)$ with a slight abuse of notation.

We propose the SPI for approximate posterior inference, where we first select knowledge $p_\theta(s|R, C)$ and then infer the response latent variable $p_\theta(z|R, C_s)$.

2.3.1. Posterior Knowledge Selection

First, we shall delve into the options for posterior knowledge selection, comparing the use of a simple uniform prior with that of a learnable prior. To infer the preferred knowledge index s, we shall sample from the posterior,

$$s \sim p_\theta(s|R, C) = \int p_\theta(s|z, R, C) p_\theta(z|R, C) dz. \tag{8}$$

Since the above integration is intractable, we approximate $p_\theta(z|R, C)$ in Eq. (8) by a point mass at the context-conditioned prior mean. Denote $\mu = f_{\alpha_2}(C_s)$. Then

$$s \sim p_\theta(s|z = \mu, R, C) \propto p_\theta(s, z = \mu|R, C). \tag{9}$$

For simplicity, we still use $p_\theta(s|R, C)$ to represent the approximate posterior distribution $p_\theta(s|z = \mu, R, C)$ and we use $p_\beta(R|C_s)$ to represent $p_\beta(R|s, C, z = \mu)$.

Uniform Prior with Top-S Initializer For posterior inference with uniform prior, we have

$$\begin{aligned}
p_\theta(s|R, C) &\propto p(s|C) p_\beta(R|C_s) \\
&\propto p_\beta(R|C_s).
\end{aligned} \tag{10}$$

In this case, the choice of the knowledge s is completely dependent on the response generation model. To be concrete, for each of the knowledge candidate with its history, C_i, $i \in$

$\{1, \ldots, M\}$, we first concatenate it with dialogue history H, the posterior logits are defined by

$$\mathbb{P}(s = i) = \frac{p_\beta(R|C_i)}{\sum_{i=1}^{M} p_\beta(R|C_i)}. \tag{11}$$

Or we can greedily choose the one that gives the best generation performance to ease the computation,

$$s = \arg\max_i p_\beta(R|C_i). \tag{12}$$

However, in the case of enormous knowledge candidates (i.e. M is large), the brutal search across all M candidates can be computationally inefficient. In this case, we propose to recruit an additional linear layer, $f_\gamma(C)$, (e.g., a classification head following BART encoder) as an initializer to narrow down the search space.

Based on the output logits from $f_\gamma(C)$, we can select top-S knowledge candidates, where $S \ll M$. Then we can leverage the aforementioned process to select knowledge by sampling from the posterior,

$$\mathbb{P}(s = i) = \frac{p_\beta(R|C_i)}{\sum_{i=1}^{S} p_\beta(R|C_i)}. \tag{13}$$

Or greedily,

$$s = \arg\max_i p_\beta(R|C_i), i \in \{1, \ldots, S\}. \tag{14}$$

This additional top-S initializer can be learned using cross-entropy loss between the predicted logits and the ground-truth label. The selection of the ground-truth label can either be derived from gold annotations, from posterior knowledge selection, or potentially a combination of both. In our experiments, we utilize both gold annotations and selected knowledge to enhance the training of the initializer.

Learnable Prior While the fixed uniform prior is straightforward, it necessitates the use of a Top-S initializer for effective operation. This leads us to contemplate if there is a way to bypass the need for an initializer, thereby enhancing the model's coherence. To this end, we employ the leanable prior,

$$s \sim p_{\alpha_1}(s|C) = \frac{\exp\left(f_{\alpha_1}(C_s)\right)}{\sum_{i=1}^{M} \exp\left(f_{\alpha_1}(C_i)\right)}, \tag{15}$$

where $f_{\alpha_1}(\cdot)$ denotes BRAT encoder and classification head. For posterior distribution, we sample from,

$$p_\theta(s|R, C) \propto p_{\alpha_1}(s|C) p_\beta(R|C_s), \tag{16}$$

or select the knowledge greedily,

$$s = \arg\max_i \exp\left(f_{\alpha_1}(C_i)\right) p_\beta(R|C_i), \tag{17}$$

where $i \in \{1, \ldots, M\}$.

The learnable prior can be updated using either gold annotations or posterior knowledge selection. Mirroring the training of initializer, we incorporate both these elements to optimize this learnable prior.

2.3.2. Posterior Inference of Response Latent Variable with Short-Run MCMC

Previous work (Rashkin et al., 2021) defines three control codes and uses them as a prefix of the inputs to indicate how the selected knowledge is presented in the gold response. It can be considered as a high-level abstraction or summary of the future response, whereas we choose a more flexible definition of abstraction as a trainable control code or a continuous prompt that is inferred from the future response given the history and selected knowledge.

After selecting the knowledge K_s, we infer the continuous response latent variable z by sampling from $p_\theta(z|R, C_s)$

$$z \sim p_\theta(z|R, C_s) \propto p_\theta(z, s|R, C) \tag{18}$$

using Langevin dynamics

$$z^{t+1} = z^t + \delta \nabla_z \log p_\theta(z^t|R, C_s) + \sqrt{2\delta}\epsilon_t, \tag{19}$$

where $\epsilon_t \sim \mathcal{N}(0, \mathbf{I})$, t indexes the time step of the Langevin dynamics and δ is the discretization step size. The gradient term is tractable since

$$\begin{aligned} &\nabla_z \log p_\theta(z|R, C_s) = \nabla_z \log p_\theta(z, R|C_s) \\ &= \nabla_z \log p_{\alpha_2}(z|C_s) + \nabla_z \log p_\beta(R|z, C_s), \end{aligned} \tag{20}$$

where $\log p_{\alpha_2}(z|C_s) = \|z - f_{\alpha_2}(C_s)\|^2/2 + \text{constant}$ and the second term is the response generation model. Both derivatives are tractable and can be computed by back-propagation.

The Langevin dynamics in Equation (19) involves a drift term (denoted by gradient) and a diffusion term. If $z^t \sim p_\theta(z^t|R, C_s)$, the drift term $\nabla_z \log p_\theta(z^t|R, C_s)$ aims to shift the distribution of z^t towards basins of high log-posterior. $p_\theta(z|R, C_s)$ can be further recovered by smoothing with the diffusion term $\sqrt{2\delta}\epsilon_t$, which induces randomness in sampling process.

However, running sufficiently long Markov chains is computationally impractical since the back-propagation through the generation model is required in each iteration according to Eq. (19). Earlier works (Pang et al., 2021a) adopt short-run MCMC (Nijkamp et al., 2019) in text modeling where they propose to approximately sample from the posterior distribution with a fixed small number of steps. Here we further scale up this idea in the scenario of PLMs. That said,

we propose the following sampling procedure,

$$z^0 \sim p_{\alpha_2}(z|C_s),$$
$$z^{t+1} = z^t + \delta \nabla_z \log p_\theta(z^t|R, C_s) + \sqrt{2\delta}\epsilon_t, \quad (21)$$

where $t = 1, \ldots, T$, and the initial state for the Markov chain is sampled from the context-conditioned prior distribution. The total length of the Markov chain is rather small (e.g. $T = 5$). Further theoretical underpinnings of this approximate sampling and learning method can be found in Appendix A.

2.4. Algorithms

The choice of prior for the discrete variable s leads to minor variations in the learning and generation algorithms.

Learning with Uniform Knowledge Prior Given learning iterations $\tau = 1, \ldots, T_L$, the generative model with parameters $\theta = \{\alpha = \alpha_2, \beta\}$ can be updated through

$$\theta_{\tau+1} = \theta_\tau + \eta_1 \Delta\theta,$$
$$\Delta\theta = \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{p_{\theta_\tau}(s^n, z^n|R^n, C^n)}[\nabla_\theta \log p_{\alpha_2}(z^n|C_s^n) + \nabla_\theta \log p_\beta(R|z^n, C_s^n)]. \quad (22)$$

The additional top-S initializer $f_\gamma(C)$ with parameters γ can be viewed as a multi-label classifier or multiple binary classifiers and be updated by cross-entropy loss,

$$\mathcal{L}_{\text{CE}}(y, C) = -\sum_{i=1}^{M} y_i \log f_\gamma(C_i) + (1 - y_i)\log(1 - f_\gamma(C_i)),$$
$$\gamma_{\tau+1} = \gamma_\tau - \eta_2 \frac{1}{N}\sum_{n=1}^{N} \nabla_\gamma \mathcal{L}_{\text{CE}}(y^n, C^n), \quad (23)$$

where $\{y^n\}_{n=1}^N$ denotes labels which can be obtained by posterior knowledge selection and/or annotations (if we use both posterior knowledge selection and annotations, it is possible that two of y_i's equal to 1). Therefore, the learned initializer can output top-S candidates that likely include the gold knowledge or the one from posterior knowledge selection.

Learning with Learnable Knowledge Prior The generative model with parameters $\theta = \{\alpha = (\alpha_1, \alpha_2), \beta\}$ can be updated through

$$\theta_{\tau+1} = \theta_\tau + \eta_1 \Delta\theta,$$
$$\Delta\theta = \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{p_{\theta_\tau}(s^n, z^n|R^n, C^n)}[\nabla_\theta \log p_{\alpha_1}(s^n|C^n) + \nabla_\theta \log p_{\alpha_2}(z^n|C_s^n) + \nabla_\theta \log p_\beta(R|z^n, C_s^n)]. \quad (24)$$

The learnable knowledge prior with parameter α_1 also functions akin to a multi-label classifier. We use both gold annotations and posterior knowledge selection as ground-truth labels, mirroring the updating of the initializer.

Generation We can get a response by greedy search as summarized in Algorithm 2. Given dialogue context $C = (H, \mathbf{K})$, we can first select the knowledge candidate with the highest logit from the initializer $f_\gamma(C)$ or learnable prior $f_{\alpha_1}(C)$ based on different prior choices,

$$s = \arg\max_i f_\gamma(C_i),\ i \in \{1, \ldots, S\}. \quad (25)$$

$$s = \arg\max_i f_{\alpha_1}(C_i),\ i \in \{1, \ldots, M\}. \quad (26)$$

Then we use sample mean of the response latent variable $p_{\alpha_2}(z|C_s) \sim \mathcal{N}(f_{\alpha_2}(C_s), \mathbf{I})$ to generate dialogue,

$$\hat{R} \sim p_\beta(R|z = f_{\alpha_2}(C_s), C_s). \quad (27)$$

Algorithm 1 Learning with Sequential Posterior Inference

input: Observed examples $\{C^n, R^n\}_{n=1}^{N_{\text{train}}}$, total training epochs T_L, learning rate η, number of candidates S in knowledge selection initializer, number of Langevin steps T, step size δ, initial weights θ_0, γ_0.
output: Updated weights θ_T, γ_T.
for $\tau = 1$ **to** T **do**
1. Draw observed examples $\{C^n, R^n\}$.
2. **Sequential Posterior Inference**
2.1 *Posterior Knowledge Selection*
`If uniform prior`
(a) Select knowledge candidates by top-S initializer.
(b) Infer s from the top-S candidates using (14).
`If learnable prior`
Infer s from the learnable prior using (17).
2.2 *Posterior Inference of Response Latent Variable*
Infer z by T-step short-run MCMC (21) with step size δ.
3. **Update Model Parameters**
Update θ and γ according to (22),(23) or (24).
end for

Algorithm 2 Knowledge-Grounded Response Generation

input: Observed examples $\{C^n\}_{n=1}^{N_{\text{test}}}$.
output: Response $\{\hat{R}^n\}_{n=1}^{N_{\text{test}}}$.
for $n = 1$ **to** N_{test} **do**
1. Draw test example C^n.
2. Select s using the initializer $f_\gamma(C^n)$ or learnable prior $f_{\alpha_1}(C^n)$ according to (25) or (26).
3. Set z as the sample mean of the prior $p_{\alpha_2}(z|C_s^n)$.
4. Generate $\hat{R}^n$ by decoder using (27).
end for

3. Experiments

3.1. Experiment Settings

Datasets We conduct our experiments on two KGD datasets, Wizard of Wikipedia (WoW) (Dinan et al., 2019) and Holl-E (Moghe et al., 2018). In WoW, dialogues are directly grounded on the knowledge sentences retrieved from Wikipedia. 22.3k dialogues with 202k turns in WoW dataset are divided into training, validation, and test subsets. Both validation and test sets consist of seen and unseen sets, where the unseen set consists of the dialogues with the unseen initial topics during the training time. For a balance between task learning and generalizability, we merge two validation sets to select the best checkpoint. The Holl-E dataset contains 9k conversations with 90k utterances about movies. Each response is obtained based on unstructured knowledge such as plots, comments, and reviews about the movie. In both datasets, the gold label for knowledge selection is provided along with each dialogue turn. More details are included in Table 8.

Implementation Details An overview of the model structure of SPI is illustrated in Figure 1. We implement SPI with both uniform and learnable knowledge prior distributions, denoted as SPI-uniform and SPI-learnable, respectively. Our training implementation is based on pre-trained BART-base (Lewis et al., 2020). In the case of SPI-uniform, the response latent prior model with parameter α_2 is instantiated with BART encoder followed by a linear layer; the Top-S initializer is instantiated with a classification head, and the response generation model with parameter β is instantiated with BART decoder. For SPI-learnable, the learnable knowledge prior model with parameter α_1 and the response latent prior model with parameter α_2 share the BART encoder. However, they differ in that the learnable prior model requires an additional classification head, and the response latent prior model requires an additional linear layer. The response generation model with parameter β is instantiated with the BART decoder. The inferred response latent variable z is concatenated with the representation of dialogue context C_s from the BART encoder on the dimension of sequence length, acting as a special token or trainable control code. BART decoder generates responses conditioned on z and C_s through the cross-attention mechanism in each Transformer layer.

Training Details We train our model with Adam optimizer with a learning rate of 1e-7 and a weight decay of 0.005. A linear scheduler is utilized to adjust the learning rate for each step. The batch size is set as 32. We train our model on NVIDIA Geforce A6000 GPU with 15 epochs and select the best checkpoint with the lowest loss on the validation set as our final model. The responses are generated using greedy search. We set S as 5 for knowledge selection initialization when using uniform prior. For Langevin dynamics, the number of Langevin steps and step size are 5 and 0.1, respectively. We discuss the training time cost with Langevin dynamics in Section 3.4.

3.2. Evaluation

Automatic Evaluation To evaluate the knowledge selection performance on both datasets, we use the accuracy (Acc) score, the ratio of the test samples where selected knowledge candidates are the same as the gold annotations. As for estimating the quality of generated responses from different models, we utilize the classical overlap-based metrics: BLEU-3 (B3), BLEU-4 (B4) (Papineni et al., 2002), Rouge-1 (R1) and Rouge-2 (R2) (Lin, 2004) to measure the distance from the golden answers. Perplexity (PPL) is the exponential negative log-likelihood of the model generating gold responses. We use distinct scores (Dist-1 and Dist-2) (Li et al., 2016) to calculate the ratio of distinct uni-gram and bi-grams at the corpus level, which reflect the diversity of generated responses.

Moreover, we adopt automatic metrics, especially for evaluating the faithfulness of the generated responses, including FeQA (Durmus et al., 2020), QuestEval (Scialom et al., 2021), and the overlap-based performance given the oracle knowledge. FeQA and QuestEval are both question-answering-based frameworks, relying on iterations of question generation based on the generated text and question answering (QA) given the context. The QA performance's accuracy is considered equivalent to the degree of faithfulness. QuestEval has two modes: (1) reference-dependent (RD) mode assesses the generated text with ground-truth references, and (2) reference-free (RF) mode conducts the assessment when no gold reference is available.

Human Evaluation For a comprehensive evaluation, we use human evaluation to compare the generated responses from our model with those from one of the previous SOTA models, KnowledGPT (Zhao et al., 2020).[1] We assess the responses quality from three aspects: *Fluency*, *Relevance*, and *Faithfulness*. *Fluency* assesses whether the response is complete, grammatically correct, and self-consistent without repetition, while *Relevance* evaluates whether the selected knowledge and the corresponding response are relevant to the dialogue history. Both fluency and relevance are assessed using A/B testing. We evaluate *Faithfulness* using a 4-point Likert scale. A faithful response should be fully supported by the dialogue context of external knowledge and history and correctly convey the information in external knowledge. 50 data samples are randomly selected from each test set, and we ensure that three annotators evaluate

[1] Human evaluation is conducted on Amazon Mechanical Turk (AMT) (https://www.mturk.com/).

MODEL	WOW SEEN								WOW UNSEEN							
	PPL↓	B3↑	B4↑	R1↑	R2↑	DIST-1↑	DIST-2↑	ACC↑	PPL↓	B3↑	B4↑	R1↑	R2↑	DIST-1↑	DIST-2↑	ACC↑
$BART_{cat}$	19.7	6.7	4.3	19.3	5.1	7.1	29.9	–	24.5	–	4.1	18.9	4.5	5.3	22.2	–
$BART_{SKT}$	20.3	7.6	4.4	19.4	5.4	6.8	30.3	26.8	22.3	–	4.6	19	4.7	5.2	24.5	18.3
$BART_{FiD}$	**9.5**	7.9	5.8	20.9	7.8	10.4	39.6	–	**10.5**	8.1	6.1	20.9	7.9	6.7	24.2	–
ZRKGC	40.4	2.8	1.8	18.6	2.4	5.4	22.5	–	41.5	18.6	1.1	18.5	2.4	3.4	15.6	–
DRD	23.0	7.5	5.5	18.0	–	–	–	–	25.6	16.5	4.3	16.5	–	–	–	–
PIPM	42.7	–	3.3	19.9	7.3	–	26.4	27.7	65.7	–	2.5	17.6	5.4	–	17.7	19.4
COLV	39.6	–	2.9	20.6	7.9	–	29.7	30.1	54.3	–	2.1	19.7	6.3	–	20.1	18.9
KAT-TSLF	14.4	9.1	6.7	21.7	7.6	9.5	38.3	–	15.8	8.3	6.0	20.7	7.2	6.7	**26.0**	–
KNOWLEDGPT	19.2	9.5	7.2	22.0	7.9	8.9	36.2	28.0	22.3	8.3	6.0	20.5	6.7	6.0	23.8	24.0
SPI-LEARNABLE	16.1	**10.2**	**7.7**	**22.7**	**8.8**	10.5	40.0	**36.5**	18.4	**9.8**	**7.4**	21.9	8.3	6.5	23.1	**34.8**
SPI-UNIFORM	17.1	**10.2**	**7.7**	**22.7**	**8.8**	**10.8**	**40.9**	36.2	19.1	9.6	7.3	**22.0**	**8.5**	**6.9**	24.3	34.6
1/2 DATA	18.2	9.7	7.3	21.8	8.1	10.6	40.6	34.3	20.1	9.2	6.9	21.1	7.7	6.5	23.0	33.2
1/4 DATA	18.7	9.3	6.9	21.6	7.8	10.1	39.0	33.6	20.7	8.9	6.6	20.9	7.3	6.3	23.1	32.5
1/8 DATA	20.3	7.9	5.7	20.2	6.7	9.4	35.8	31.4	22.0	8.1	6.0	19.6	6.5	5.8	20.7	30.6
1/16 DATA	22.0	7.0	4.9	18.7	5.6	8.9	34.0	27.5	23.6	7.2	5.2	18.5	5.7	5.7	20.8	27.0

Table 1. Automatic evaluation results on WoW test sets. PPL is short for Perplexity; B3 and B4 represent BLEU-3 and BLEU-4; R1 and R2 denote Rouge-1 and Rouge-2; Dist-1 and Dist-2 denote uni-gram and bi-gram distinct metrics. Numbers of previous models are taken from (Zhao et al., 2019; Li et al., 2020; Chen et al., 2020; Zhan et al., 2021; Zhao et al., 2020; Liu et al., 2021). SPI achieves new SOTA performance on WoW tet sets. The performance of our proposed model under the low-resource settings is shown in the last four rows.

MODEL	ORACLE PERFORMANCE					FEQA	QUESTEVAL	
	PPL↓	B3	B4	R1	R2		RD	RF
WoW Seen								
KNOWLEDGPT	9.1	19.2	15.5	34.5	17.3	48.1	42.2	43.5
SPI-LEARNABLE	8.9	19.3	15.7	34.6	17.5	48.3	**45.1**	**46.6**
SPI-UNIFORM	**8.7**	**20.0**	**16.3**	**36.1**	**18.7**	**49.2**	44.4	46.0
WoW Unseen								
KNOWLEDGPT	9.8	18.3	14.6	33.8	16.5	47.4	41.0	42.2
SPI-LEARNABLE	9.5	19.2	15.5	34.0	17.2	48.1	**44.2**	**45.7**
SPI-UNIFORM	**9.2**	**20.1**	**16.3**	**36.0**	**18.7**	**49.6**	44.0	**45.7**

Table 2. The results on automatic faithfulness metrics on WoW test sets. The proposed model, SPI, consistently outperforms KnowledGPT on all the metrics, showing its superior faithfulness.

MODEL	PPL↓	B4	R1	R2	DIST-2	ACC
SKT	48.9	-	29.8	23.1	-	29.2
DUKENET	42.7	19.2	32.6	19.6	28.5	30.4
PIPM	39.2	18.3	30.8	24.0	27.2	30.7
COLV	34.8	20.3	32.0	25.8	29.9	32.7
SPI-UNIFORM	**12.6**	**30.7**	**38.3**	**31.7**	**30.6**	**38.3**

Table 3. Automatic evaluation results on Holl-E test set. Numbers of previous models are taken from (Kim et al., 2019; Meng et al., 2020; Chen et al., 2020; Zhan et al., 2021). Our model outperforms all the strong baselines and achieves new SOTA performance.

each sample. Further details and annotator instructions are included in Appendix E.

3.3. Results

Table 1 and Table 3 report automatic evaluation results of our proposed model on WoW and Holl-E test sets. We compare our model with a number of previous strong models on both datasets and highlight the best performance of each metric in bold. The baseline models are introduced in Appendix C. Comparing SPI models that learn with two prior hypotheses, SPI with uniform knowledge prior shows comparable performance on the overall response generation performance and knowledge selection accuracy, but it ensures better diversity of the generated response. Our proposed method achieves new SOTA performance on both datasets. It outperforms all the previous strong baseline models on knowledge selection accuracy and overlap-based metrics, indicating a higher quality of knowledge selection and response generation. Comparing SPI with uniform knowledge prior and KnowledGPT, our model shows a 11.4% on Rouge-2, and 29.3% on accuracy on the WoW test seen set. Meanwhile, the improvements on the WoW test unseen set are even larger. This proves the better *generalizability* of our model. The improvements on the Holl-E dataset are at least 17% for all the metrics except Distinct-2 (2%).

Furthermore, SPI models consistently outperform KnowledGPT on all the automatic faithfulness metrics in Table 2, showing its superior faithfulness. Our advantage over other models in distinct scores (Table 1 and Table 3) also shows that our model tends to generate more diverse responses, especially in the seen domain. In WoW unseen set, SPI underperforms KAT-TSLF (Liu et al., 2021) on the Distinct-2 metric. KAT-TSLF proposes a BART-based model pre-trained on a large dialogue corpus with pseudo-knowledge pairs and then adapted to WoW dataset through fine-tuning. Regarding its performance on other metrics, we believe pre-training is the major contributor to diversity. Our model achieves the second-best performance on Distinct-2 with no additional pre-training step or data resource.

PPL scores of our model are less satisfying than the deterministic models, i.e. BART-based FiD (Izacard & Grave, 2021). However, it is necessary to emphasize that even though there is a correlation between PPL and human evalu-

MODEL	FLUENCY		RELEVANCE		FAITHFULNESS	
	SEEN	UN.	SEEN	UN.	SEEN	UN.
KNOWLEDGPT	62.5%	60.3%	70.8%	62.2%	3.33	3.42
SPI-UNIFORM	**88.7%**	**83.3%**	79.8%	**74.4%**	**3.66**	**3.65**

Table 4. Human evaluation results on WoW test sets, in terms of *Fluency*, *Relevance*, and *Faithfulness*. *Un.* is short for the unseen set. A pairwise t-test is conducted to validate the significance of the improvements, and the corresponding results in bold are significantly better than those from the baseline model ($p < 0.05$).

ation to some extent, it is not directly reflecting the quality of response generation when the PPL is low because of the likelihood trap confirmed in (Zhang et al., 2021). [2]

Table 4 lists the human evaluation results on both test sets of WoW, comparing KnowledGPT and SPI with uniform prior in terms of *Fluency*, *Relevance*, and *Faithfulness*. The details about how scores are calculated are stated in Appendix E. A pairwise individual t-test validates the significance of the advantages of our model over KnowledGPT. Our model is more likely to generate fluent responses, select more relevant knowledge, and ensure coherence to the dialogue history. According to the criteria of Faithfulness evaluation, both KnowledGPT and our model generate partially faithful responses. Nevertheless, our model generates significantly more faithful responses, while enhancing diversity given the Distinct scores in Table 1. Moreover, a case study is also included in Appendix D.

3.4. Ablation Study

Low-resource settings Our model demonstrates high training efficiency under low-resource settings. We train our model using the same hyper-parameter settings as SPI-uniform with $1/2$, $1/4$, $1/8$, and $1/16$ of data samples on WoW datasets. From Table 1, with the increasing number of training data samples, the performance of all the metrics improves consistently. With only $1/4$ data samples, our model can still perform comparably or even better than that of other strong baseline models with full data resources. We compare our performance with KAT-TSLF under low-resource settings, as shown in Table 7. SPI with uniform knowledge prior appears to drop less on the performance under $1/4$ and $1/8$ data settings, with much less training cost than KAT-TSLF. KAT-TSLF relies on pre-training with a large dialogue corpus to prevent the model from poor performance under the low-resource setting. Because of pre-training, KAT-TSLF shows zero-shot KGD ability and gets better diversity in some low-resource settings. However, we find no difficulty applying SPI for pre-training.

[2] If the PPL of the model is too low, the correlation with human judgment decreases.

Impact of top-S selection When learning with uniform knowledge prior, the choice of S is an essential hyper-parameter. To study the impact of it, we conduct experiments when $S = 1/3/5/10$ with all the other settings kept the same. As the results listed in Table 5, when the initializer is only optimized on gold labels for knowledge selection without posterior knowledge selection ($S = 1$), the model performs the best knowledge selection accuracy. However, with more knowledge candidates produced by the initializer, the diversity of generated responses is on the rise, whereas the best overlap-based accuracy achieves with top-5 knowledge candidates. It shows that injecting posterior information into the initializer during training improves the faithfulness (FeQA and QuestEval scores) of the generated responses. This verifies our assumption about the inherent correlation between knowledge selection and response generation. It also proves that better knowledge selection helps with better results but does not guarantee better responses because the generation can still hallucinate and deviate from the knowledge source provided. The two paradigms of KGD tasks should be optimized jointly.

Impact of the number of Langevin steps In Table 6, we further study the impact of the number of Langevin steps on response generation. When training these models, all the experimental settings except the number of Langevin steps are kept the same as SPI with uniform knowledge prior. When no Langevin step is taken, the response latent variable z degenerates to be a deterministic representation. Posterior inference of z further boosts the performance of the SPI model on overlap-based accuracies, demonstrating the effectiveness of the proposed method, especially in the unseen domain. It also improves both diversity and faithfulness by providing a high-level abstraction of the further response with the response latent variable. Posterior inference with Langevin dynamics requires the model to use MCMC, which sequentially queries the BART decoder to obtain the gradient from the generator for updating the response latent variable z. One possible concern is the increasing training cost when more Langevin steps are taken. We calculate the training time per epoch for models with different Langevin steps. Posterior inference of response latent variable z with Langevin steps to be five only extends the training time per epoch by 5.1%, which does not bring much burden on the training process.

4. Related Work

Knowledge-Grounded Dialogue Generation KGD task has been investigated for many years (Dinan et al., 2019; Feng et al., 2021). Due to one-to-many problems in knowledge selection, one line of existing work adopts variational inference based methods, which construct a latent variable for knowledge selection and optimize it with variational

TOP-S	WOW SEEN						WOW UNSEEN					
	B-4	R-2	DIST-2	FEQA	Q.E.(RD/RF)	ACC	B-4	R-2	DIST-2	FEQA	Q.E.(RD/RF)	ACC
1	7.3	8.4	36.6	40.4	41.1/43.0	**37.0**	6.9	7.7	22.5	39.2	39.9/41.8	**34.7**
3	7.4	8.3	39.4	40.7	41.4/43.2	34.1	7.0	7.8	22.5	40.5	40.5/42.2	32.2
5 (OURS)	**7.7**	**8.8**	40.9	**49.2**	**44.4/46.0**	36.2	**7.3**	**8.5**	24.3	**49.6**	**44.0/45.7**	34.6
10	7.2	8.8	**41.1**	48.0	42.4/44.2	36.4	7.3	8.4	**24.4**	47.7	42.3/44.0	34.6

Table 5. Ablation study on the impact of the choice of top-S for posterior knowledge selection initialization on WoW test sets. Q.E. is short for QUESTEVAL. Our final model with top-5 knowledge candidates shows a balance between diversity and overlap-based accuracy on the quality of generated responses.

LANGEVIN STEPS	WOW SEEN					WOW UNSEEN					TR. TIME (/EPOCH)
	B4	R2	DIST-2	FEQA	Q.E.(RD/RF)	B4	R2	DIST-2	FEQA	Q.E.(RD/RF)	
0	7.4	8.7	40.3	47.4	43.8/45.6	6.9	8.2	23.5	48.0	42.9/44.6	3.50HRS
1	7.6	8.7	40.3	47.9	44.2/45.9	**7.4**	8.4	23.1	47.9	43.5/45.1	3.56HRS
5 (OURS)	**7.7**	**8.8**	**40.9**	**49.2**	**44.4/46.0**	7.3	**8.5**	**24.3**	**49.6**	**44.0/45.7**	3.68HRS

Table 6. Ablation study on the impact of the number of Langevin steps on WoW test sets. Q.E. is short for QUESTEVAL. We also present the training time (Tr. Time) per epoch under each setting. As the number of Langevin steps increases, the performance on the test seen set consistently improves, while the training time cost also increases slightly.

inference (Lian et al.; Kim et al., 2019; Li et al., 2020; Chen et al., 2020). Further explorations extend the formulation to two collaborative latent variables to augment response generation or enhance knowledge selection. (Zhan et al., 2021) utilizes two collaborative latent variables to model the distributions of knowledge and response simultaneously, while (Fu et al., 2022) introduces two latent variables to indicate the fragment of personal memory to evoke and the knowledge candidate to select, respectively. Another line of research bypasses the knowledge selection step but relies on improving knowledge usage during response generation given all the knowledge sentences (Zhao et al., 2020; Liu et al., 2021). Since PLMs hallucination problem (Ji et al., 2022a) leads to some of the challenges in faithfulness, we note that to reduce hallucination in KGD systems, existing work focuses on guiding the model on correct knowledge usage (Rashkin et al., 2021; Ji et al., 2022b) or providing dialogue models with better knowledge augmentation by improving knowledge selection performance (Shuster et al., 2021). In this work, SPI jointly improves both processes and shows a significant faithfulness advantage through automatic and human evaluation.

Posterior Inference (Han et al., 2017) proposes to learn generative image models by alternating back-propagation, which first infers the latent variable by sampling from its posterior distribution and then updates the model parameters by usual back-propagation. Our SPI shares the same insight. To sample efficiently in the continuous latent space, (Tieleman, 2008; Nijkamp et al., 2019) propose different versions of MCMC to learn the generative models. Specifically, short-run MCMC (Nijkamp et al., 2019) proposes finite-step inference dynamics guided by an energy-based model. We further scale up this idea in the scenarios of PLMs to sample from the continuous latent space.

5. Conclusion

In this work, we propose a probabilistic model with dual latent variables, one discrete latent variable for knowledge selection and one continuous latent variable for response generation. This model is effectively optimized by approximate MLE with the proposed posterior inference method, SPI. Our model has demonstrated its validity and superiority with both theoretical analysis and empirical studies. Further ablation studies show that SPI can search the discrete and continuous spaces efficiently by our proposed initializer and short-run MCMC in fine-tuning PLMs. We also find that faithfulness and diversity are emergent properties that can be improved while enhancing the inherent correlation with knowledge selection and response generation and providing the generator with a high-level abstraction of the future response. Although in this paper, we mainly focus on KGD scenarios, our proposed method, SPI, has the potential to be applied to other knowledge-intensive tasks which require reasoning ability during text generation. We leave further exploration to future work.

Acknowledgements

Y. N. Wu was partially supported by NSF DMS-2015577. P. Fung was partially supported by the HKJCCT21EG01 of the Hong Kong Jockey Club. We would like to thank the five anonymous reviewers for their constructive comments.

References

Chen, X., Meng, F., Li, P., Chen, F., Xu, S., Xu, B., and Zhou, J. Bridging the gap between prior and posterior knowledge selection for knowledge-grounded dialogue generation. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 3426–3437, 2020.

Cremer, C., Li, X., and Duvenaud, D. Inference suboptimality in variational autoencoders. In *International Conference on Machine Learning*, pp. 1078–1086. PMLR, 2018.

Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., and Weston, J. Wizard of wikipedia: Knowledge-powered conversational agents. In *International Conference on Learning Representations*, 2018.

Dinan, E., Roller, S., Shuster, K., Fan, A., Auli, M., and Weston, J. Wizard of Wikipedia: Knowledge-powered conversational agents. In *Proceedings of the International Conference on Learning Representations (ICLR)*, 2019.

Durmus, E., He, H., and Diab, M. Feqa: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 5055–5070, 2020.

Feng, S., Patel, S. S., Wan, H., and Joshi, S. Multidoc2dial: Modeling dialogues grounded in multiple documents. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6162–6176, 2021.

Fu, T., Zhao, X., Tao, C., Wen, J.-R., and Yan, R. There are a thousand hamlets in a thousand people's eyes: Enhancing knowledge-grounded dialogue with personal memory. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 3901–3913, 2022.

Ghazvininejad, M., Brockett, C., Chang, M.-W., Dolan, B., Gao, J., Yih, W.-t., and Galley, M. A knowledge-grounded neural conversation model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Han, T., Lu, Y., Zhu, S.-C., and Wu, Y. N. Alternating back-propagation for generator network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31, 2017.

Izacard, G. and Grave, É. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pp. 874–880, 2021.

Ji, Z., Lee, N., Frieske, R., Yu, T., Su, D., Xu, Y., Ishii, E., Bang, Y., Madotto, A., and Fung, P. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 2022a.

Ji, Z., Liu, Z., Lee, N., Yu, T., Wilie, B., Zeng, M., and Fung, P. Rho: Reducing hallucination in open-domain dialogues with knowledge grounding. *arXiv preprint arXiv:2212.01588*, 2022b.

Kim, B., Ahn, J., and Kim, G. Sequential latent knowledge selection for knowledge-grounded dialogue. In *International Conference on Learning Representations*, 2019.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In Bengio, Y. and LeCun, Y. (eds.), *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014. URL http://arxiv.org/abs/1312.6114.

Langevin, P. *On the theory of Brownian motion*. 1908.

Lewis, M., Liu, Y., Goyal, N., Ghazvininejad, M., Mohamed, A., Levy, O., Stoyanov, V., and Zettlemoyer, L. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *ACL 2020*, 2020.

Li, H. and Han, T. Learning sparse latent representations for generator model. *arXiv preprint arXiv:2209.09949*, 2022.

Li, J., Galley, M., Brockett, C., Gao, J., and Dolan, W. B. 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*, pp. 110–119, 2016.

Li, L., Xu, C., Wu, W., Zhao, Y., Zhao, X., and Tao, C. Zero-resource knowledge-grounded dialogue generation. *Advances in Neural Information Processing Systems*, 33: 8475–8485, 2020.

Lian, R., Xie, M., Wang, F., Peng, J., and Wu, H. Learning to select knowledge for response generation in dialog systems.

Lin, C.-Y. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pp. 74–81, 2004.

Liu, S., Zhao, X., Li, B., Ren, F., Zhang, L., and Yin, S. A three-stage learning framework for low-resource knowledge-grounded dialogue generation. *arXiv preprint arXiv:2109.04096*, 2021.

Meng, C., Ren, P., Chen, Z., Sun, W., Ren, Z., Tu, Z., and Rijke, M. d. Dukenet: A dual knowledge interaction network for knowledge-grounded conversation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 1151–1160, 2020.

Moghe, N., Arora, S., Banerjee, S., and Khapra, M. M. Towards exploiting background knowledge for building conversation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pp. 2322–2332, Brussels, Belgium, October-November 2018. Association for Computational Linguistics. doi: 10.18653/v1/D18-1255. URL https://aclanthology.org/D18-1255.

Nijkamp, E., Hill, M., Zhu, S.-C., and Wu, Y. N. Learning non-convergent non-persistent short-run mcmc toward energy-based model. *Advances in Neural Information Processing Systems*, 32, 2019.

OpenAI. Chatgpt: Optimizing language models for dialogue, Jan 2023. URL https://openai.com/blog/chatgpt/.

Pang, B., Han, T., Nijkamp, E., Zhu, S.-C., and Wu, Y. N. Learning latent space energy-based prior model. In Larochelle, H., Ranzato, M., Hadsell, R., Balcan, M., and Lin, H. (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 21994–22008. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/file/fa3060edb66e6ff4507886f9912e1ab9-Paper.pdf.

Pang, B., Nijkamp, E., Han, T., and Wu, Y. N. Generative text modeling through short run inference. *arXiv preprint arXiv:2106.02513*, 2021a.

Pang, B., Zhao, T., Xie, X., and Wu, Y. N. Trajectory prediction with latent belief energy-based model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11814–11824, June 2021b.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pp. 311–318, 2002.

Rashkin, H., Reitter, D., Tomar, G. S., and Das, D. Increasing faithfulness in knowledge-grounded dialogue with controllable features. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 704–718, 2021.

Robbins, H. and Monro, S. A stochastic approximation method. In *Herbert Robbins Selected Papers*, pp. 102–109. Springer, 1985.

Roller, S., Dinan, E., Goyal, N., Ju, D., Williamson, M., Liu, Y., Xu, J., Ott, M., Smith, E. M., Boureau, Y.-L., et al. Recipes for building an open-domain chatbot. In *EACL*, 2021.

Scialom, T., Dray, P.-A., Lamprier, S., Piwowarski, B., Staiano, J., Wang, A., and Gallinari, P. Questeval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 6594–6604, 2021.

Serban, I., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 30, 2016.

Shuster, K., Poff, S., Chen, M., Kiela, D., and Weston, J. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pp. 3784–3803, 2021.

Tieleman, T. Training restricted boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pp. 1064–1071, 2008.

Xie, J., Zhu, Y., Li, J., and Li, P. A tale of two flows: cooperative learning of langevin flow and normalizing flow toward energy-based model. *arXiv preprint arXiv:2205.06924*, 2022.

Xu, Y., Ishii, E., Winata, G. I., Lin, Z., Madotto, A., Liu, Z., Xu, P., and Fung, P. Caire in dialdoc21: Data augmentation for information seeking dialogue system. In *Proceedings of the 1st Workshop on Document-grounded Dialogue and Conversational Question Answering (DialDoc 2021)*, pp. 46–51, 2021.

Xu, Y., Ishii, E., Cahyawijaya, S., Liu, Z., Winata, G. I., Madotto, A., Su, D., and Fung, P. Retrieval-free knowledge-grounded dialogue response generation with adapters. In *Proceedings of the Second DialDoc Workshop on Document-grounded Dialogue and Conversational Question Answering*, pp. 93–107, 2022.

Yang, C., Lin, Z., Li, J., Meng, F., Wang, W., Wang, L., and Zhou, J. Take: Topic-shift aware knowledge selection for dialogue generation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pp. 253–265, 2022.

Zhan, H., Shen, L., Chen, H., and Zhang, H. Colv: A collaborative latent variable model for knowledge-grounded dialogue generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pp. 2250–2261, 2021.

Zhang, H., Duckworth, D., Ippolito, D., and Neelakantan, A. Trading off diversity and quality in natural language generation. In *Proceedings of the Workshop on Human Evaluation of NLP Systems (HumEval)*, pp. 25–33, 2021.

Zhang, Y., Sun, S., Galley, M., Chen, Y.-C., Brockett, C., Gao, X., Gao, J., Liu, J., and Dolan, W. B. 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*, pp. 270–278, 2020.

Zhao, X., Wu, W., Tao, C., Xu, C., Zhao, D., and Yan, R. Low-resource knowledge-grounded dialogue generation. In *International Conference on Learning Representations*, 2019.

Zhao, X., Wu, W., Xu, C., Tao, C., Zhao, D., and Yan, R. Knowledge-grounded dialogue generation with pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 3377–3390, 2020.

A. Theoretical Understanding

In Section 2, we sample from $p_\theta(s, z|C, R)$ approximately. Let $q_\theta(s, z|C, R)$ be the actual distribution of the sampled (s, z).

Given model parameters θ_τ at training iteration τ, the updating rule using the approximate posterior distribution of (s, z) is one-step gradient ascent on the following function,

$$Q(\theta) = \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{q_{\theta_\tau}(s^n, z^n|R^n, C^n)}[\log p_\theta(s^n, z^n, R^n|C^n)]. \tag{28}$$

Comparing to the log-likelihood in Eq. (5), we have,

$$\begin{aligned} Q(\theta) &= L(\theta) + \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{q_{\theta_\tau}(s^n, z^n|R^n, C^n)}[\log p_\theta(s^n, z^n|R^n, C^n)] \\ &= L(\theta) - \frac{1}{N}\sum_{n=1}^{N} \mathbb{D}_{\mathrm{KL}}(q_{\theta_\tau}(s^n, z^n|R^n, C^n)||p_\theta(s^n, z^n|R^n, C^n)) \\ &+ \frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{q_{\theta_\tau}(s^n, z^n|R^n, C^n)}[\log q_{\theta_\tau}(s^n, z^n|R^n, C^n)]. \end{aligned} \tag{29}$$

With θ_τ fixed, the above equation becomes a function of θ. Then the updating rule follows the stochastic gradient of

$$\tilde{Q}(\theta) = L(\theta) - \frac{1}{N}\sum_{n=1}^{N} \mathbb{D}_{\mathrm{KL}}(q_{\theta_\tau}(s^n, z^n|R^n, C^n)||p_\theta(s^n, z^n|R^n, C^n)), \tag{30}$$

which can be viewed as a perturbation or variational lower bound of $L(\theta)$.

The fixed point of the learning algorithm that updates θ in Eq. (22) solves the following estimating equation:

$$\frac{1}{N}\sum_{n=1}^{N} \mathbb{E}_{q_{\theta(s^n, z^n|R^n, C^n)}}\left[\nabla_\theta \log p_\theta(s^n, z^n, R^n|C^n)\right] = 0. \tag{31}$$

The Monte Carlo approximation of the above expectation leads to the Robbins-Monro algorithm for stochastic approximation (Robbins & Monro, 1985). The convergence to the fixed point follows the regular conditions of the Robbins-Monro algorithm.

B. Technical Challenges of Langevin Dynamics in PLMs

The technical challenges of Langevin Dynamics in PLMs stem from the difficulties of leveraging latent variable models (LVM) in PLMs. We propose to infer the posterior distribution of discrete and continuous latent variables. For continuous latent variables, we mainly face two major difficulties: (1) The choice of hyper-parameters: step size and total number of steps. The step size determines the induced values for the drift and diffusion terms in Eq. (21). The total number of steps determines the total number of times we need to back-propagate through the BART decoder (generation model) using PyTorch auto-differentiation to calculate the gradient manually. Empirically, we find that the training process is more stable when the step size is smaller than 0.1 and the total number of steps is around five in our settings. (2) The initial distribution of the Markov chain is crucial. (Pang et al., 2021a) uses noise-initialized Markov chains for text generation. However, we find that noise initialization does not work well in PLMs. Ideally, we should run an infinitely long chain until convergence so that the final state of the Markov Chain is independent of the initial point. In the short-run case, however, the target distribution depends on the starting point (Nijkamp et al., 2019). In our case, we start from the prior distribution directly. We also employ other engineering tricks, such as gradient clamping, which can be found in our released code.

C. Baseline Models

In this work, we compare the performance of our model with nine other strong baseline models on two KGD benchmarks. We introduce each of them below:

MODEL	WOW SEEN							WOW UNSEEN						
	PPL↓	B3↑	B4↑	R1↑	R2↑	DIST-1↑	DIST-2↑	PPL↓	B3↑	B4↑	R1↑	R2↑	DIST-1↑	DIST-2↑
DRD	23.0	7.5	5.5	18.0	–	–	–	25.6	6.2	4.3	16.5	–	–	–
1/2 DATA	25.3	7.3	5.3	17.5	–	–	–	27.7	6.4	4.5	16.7	–	–	–
1/4 DATA	29.2	6.4	4.4	16.9	–	–	–	32.4	6.0	4.1	16.2	–	–	–
1/8 DATA	33.5	5.9	3.9	16.3	–	–	–	35.8	5.4	3.5	16.0	–	–	–
1/16 DATA	38.6	5.2	3.3	15.7	–	–	–	41.0	5.0	3.2	15.3	–	–	–
KAT-TSLF	**14.4**	9.1	6.7	21.7	7.6	9.5	38.3	**15.8**	8.3	6.0	20.7	7.2	6.7	**26.0**
1/4 DATA	17.6	7.7	5.5	20.3	6.8	9.9	39.1	18.4	7.5	5.2	19.9	6.4	6.6	25.1
1/8 DATA	18.8	7.1	4.9	19.8	6.3	9.9	39.5	20.1	7	4.8	19.0	5.9	6.6	25.3
ZERO DATA	100+	4.0	2.2	14.7	3	7.5	33.9	100+	4.7	2.7	14.9	3	5.7	26.4
SPI (TOP-S, OURS)	17.1	**10.2**	**7.7**	**22.7**	**8.8**	**10.8**	**40.9**	19.1	**9.6**	**7.3**	**22.0**	**8.5**	**6.9**	24.3
1/2 DATA	18.2	9.7	7.3	21.8	8.1	10.6	40.6	20.1	9.2	6.9	21.1	7.7	6.5	23.0
1/4 DATA	18.7	9.3	6.9	21.6	7.8	10.1	39.0	20.7	8.9	6.6	20.9	7.3	6.3	23.1
1/8 DATA	20.3	7.9	5.7	20.2	6.7	9.4	35.8	22.0	8.1	6.0	19.6	6.5	5.8	20.7
1/16 DATA	22.0	7.0	4.9	18.7	5.6	8.9	34.0	23.6	7.2	5.2	18.5	5.7	5.7	20.8

Table 7. Automatic evaluation results on WoW test sets under low-resource settings, compared with DRD (Zhao et al., 2019) and KAT-TSLF (Liu et al., 2021). PPL is short for Perplexity; B3 and B4 represent BLEU-3 and BLEU-4; R1 and R2 denote Rouge-1 and Rouge-2; Dist-1 and Dist-2 denote uni-gram and bi-gram distinct metrics.

SKT SKT (Kim et al., 2019) is a sequential latent knowledge selection model for multi-turn KGD tasks. Both prior and posterior distributions for knowledge selection are considered sequential processes. The model can keep track of prior and posterior distributions over knowledge, where both distributions are sequentially updated considering the responses in previous turns. We adopt the knowledge candidate selected by SKT to BART for response generation.

FiD Fusion-in-Deocder (FiD) (Izacard & Grave, 2021) is a simple yet effective model for general knowledge-intensive tasks when the context should be augmented with multiple extra documents. The model can be applied to any encoder-decoder-based PLMs. It encodes different context-document pairs in parallel and concatenates all the output hidden states together so that the decoder can attend to all these representations during generation. In this work, we use BART-base as the backbone model for a fair comparison.

DRD DRD (Zhao et al., 2019) proposes a disentangled response decoder in order to isolate parameters for generating responses that depend on dialogue context, knowledge inputs, or responses themselves. When generating one token, the decoder needs to do inference with three groups of parameters respectively and decides the final output token with a decoding manager. The proposed model is also pre-trained on the same dialogue corpus as (Liu et al., 2021), thus it is also able to perform KGD generation under a low-resource setting (Table 7).

ZRKGC ZRKGC (Li et al., 2020) focuses on the situation when the real knowledge-grounded dialogue data are not available during the training process. Two latent variables that represent the knowledge for grounding and the rate of grounding are introduced to the model. The generation process is then formalized within a probabilistic framework and optimized via variational inference.

PIPM PIPM is short for "SKT+PIPM+KDBTS" (Chen et al., 2020). The authors propose posterior information prediction and knowledge distillation-based training strategy for knowledge selection. KL divergence is leveraged to bridge the gap between prior and posterior knowledge selection.

DukeNet DukeNet (Meng et al., 2020) explicitly models knowledge tracking and knowledge shifting and formulating their interactions as dual learning without extra external supervision.

CoLV CoLV (Zhan et al., 2021) is a Collaborative Latent Variable model. Similar to our model, it also simultaneously improves the diversity of both knowledge selection and knowledge-aware response generation. However, the model still depends on variational inference for building both latent spaces.

	Wizard of Wikipedia				Holl-E		
	Train	Valid	Test Seen	Test Unseen	Train	Valid	Test
# Conversations	18,430	1,948	965	968	7,228	930	913
Avg. # Knowledge Sentences			60			58	
Avg. # Turns			9			5	

Table 8. Data statstics of Wizard of Wikipedia and Holl-E datasets.

	Topic	Chevrolet Corvette
Dialogue History	User	What do you know about the Chevrolet Corvette?
	System	The Chevy Corvette, or "vette" as it is known, is an iconic American sports car that has been produced for half a century.
	User	Do you remember the prince song Little Red Corvette ?
Selected Knowledge	KnowledGPT	Chevrolet Corvette : The first model, a convertible, was introduced at the GM Motorama in 1953 as a concept show car.
	SPI (Ours)	Little Red Corvette: "Little Red Corvette" is a song by American musician Prince.
Response	KnowledGPT	Yes, it was first introduced at the GM Motorama in 1953 as a concept show car.
	SPI (Ours)	I do. It was a song by American musician Prince.

Table 9. One case from test seen set of WoW, comparing the generated response from SPI with that from KnowledGPT.

KnowledGPT KnowledGPT (Zhao et al., 2020), as one of the previous SOTA models, equips response generation with a sequential knowledge selector and jointly optimizes both the knowledge selector and the response generator with reinforcement learning and curriculum learning. The knowledge selector first ranks all the knowledge candidates and then knowledge candidates are concatenated with dialogue history as inputs and truncated to meet the length constraint of the GPT-2.

KAT-TSLF KAT-TSLF (Liu et al., 2021) is also one of the previous SOTA models. The authors propose a three-stage learning framework for low-resource knowledge-grounded dialogue tasks. First, the dialogue history encoder and knowledge encoder are pre-trained on the dialogue corpus and knowledge base respectively. Then, the method matches each dialogue turn in the dialogue corpus with a pseudo gold knowledge from the knowledge base and use the processed new corpus to pre-train the whole model. After two-stage pre-training, the model is adapted to downstream KGD benchmarks and maintains strong performance under low-resource settings. Instead of selecting the knowledge from knowledge candidates, all the provided knowledge sentences are used as inputs, and the decoder is trained to select from all the information.

D. Case Study

Table 9 demonstrates two typical cases from WoW test sets, comparing SPI with KnowledGPT. In the presented case, the dialogue history shows a topic shift from "Chevrolet Corvette" to the "Little Red Corvette" song. However, KnowledGPT fails to capture this shift, whereas our model selects the most relevant knowledge.

E. Human Evaluation

In addition to the automatic evaluation, we conduct human evaluation to assess the quality of responses generated by our model *SPI* and baseline *KnowledGPT* on WoW. We randomly select 50 samples from each model, and each sample is evaluated by three different annotators. For each comparison, the same context, and two generated responses from each model are shown to the annotators. We require the annotators to be masters with the following qualifications: the numbers of Human Intelligence Tasks (HITs) approved are greater than or equal to 5000, and their HIT approval rates are greater than or equal to 95%. The locations of annotators are restricted to Australia, Canada, the United Kingdom, and the United States. After collecting annotations for Amazon Mechanical Turk (AMT), we calculate each score as follows: For A/B testing on Fluency and Relevance, we give one score to the model if it generates an equally good or better response than the

other one. We present the ratio of the number of data samples with one score over all the test samples. For 4-point Likert scale on Faithfulness, we assign different levels of faithfulness scores from four to one, then present the average score over all the test samples. Figure 2, Figure 3, and Figure 4 display the annotator instructions of AMT for *Fluency*, *Relevance*, and *Faithfulness*, respectively. Please find the instructions and examples for annotators in these figures.

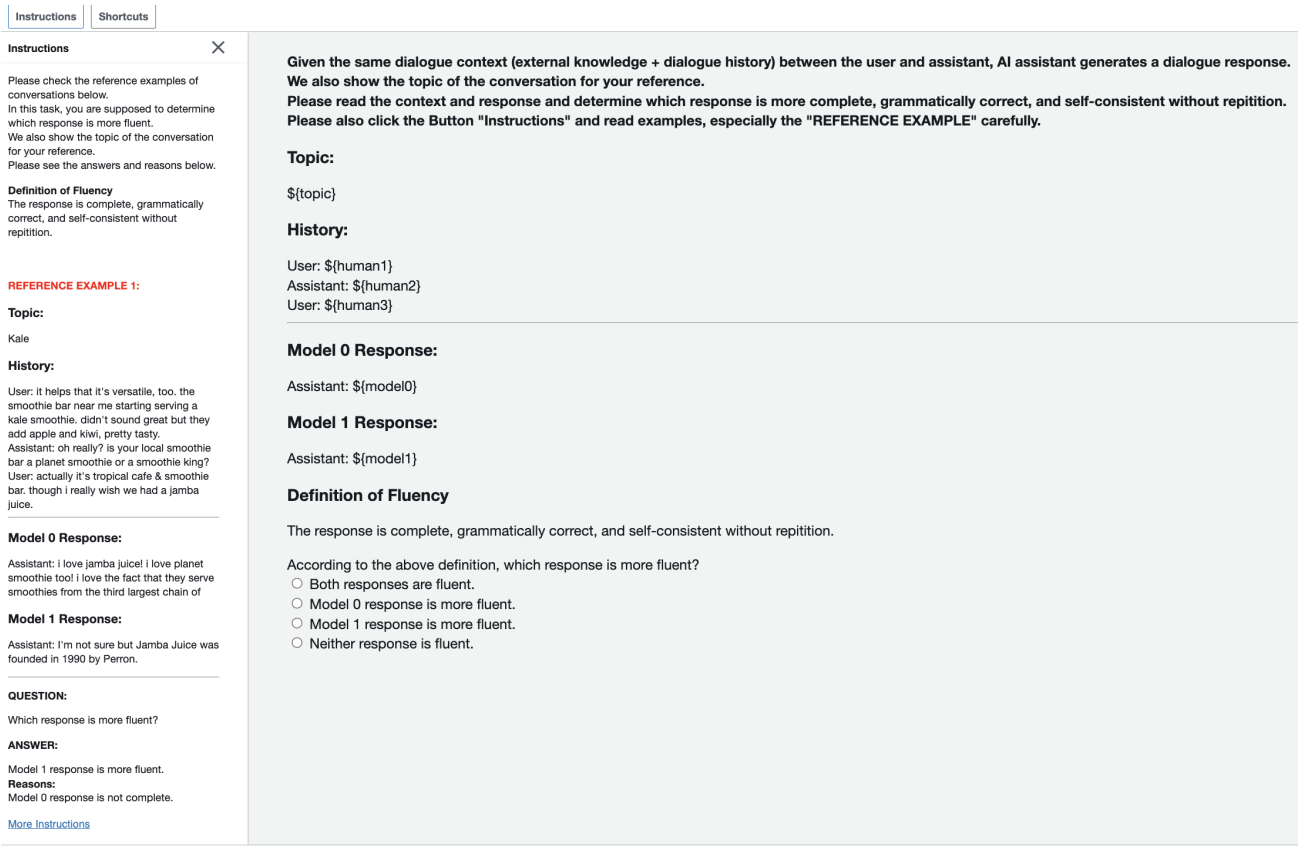

Figure 2. The annotator instruction for human evaluation on fluency via A/B testing.

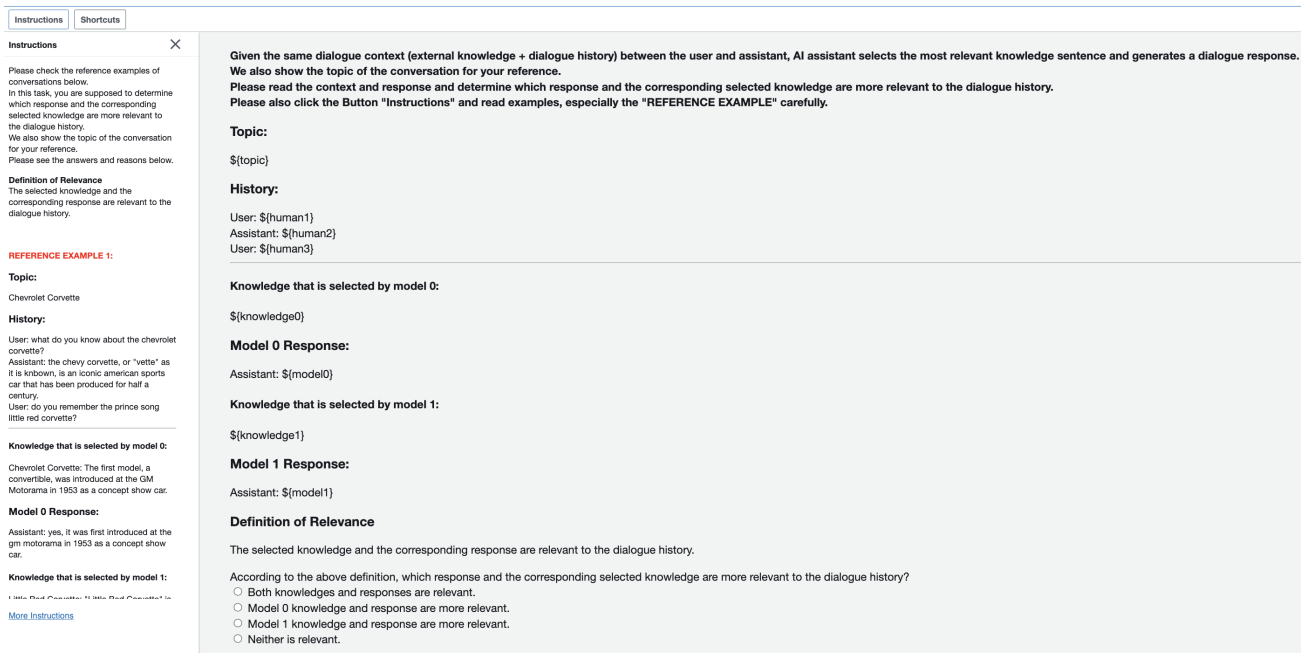

Figure 3. The annotator instructions for human evaluation on knowledge relevance via A/B testing.

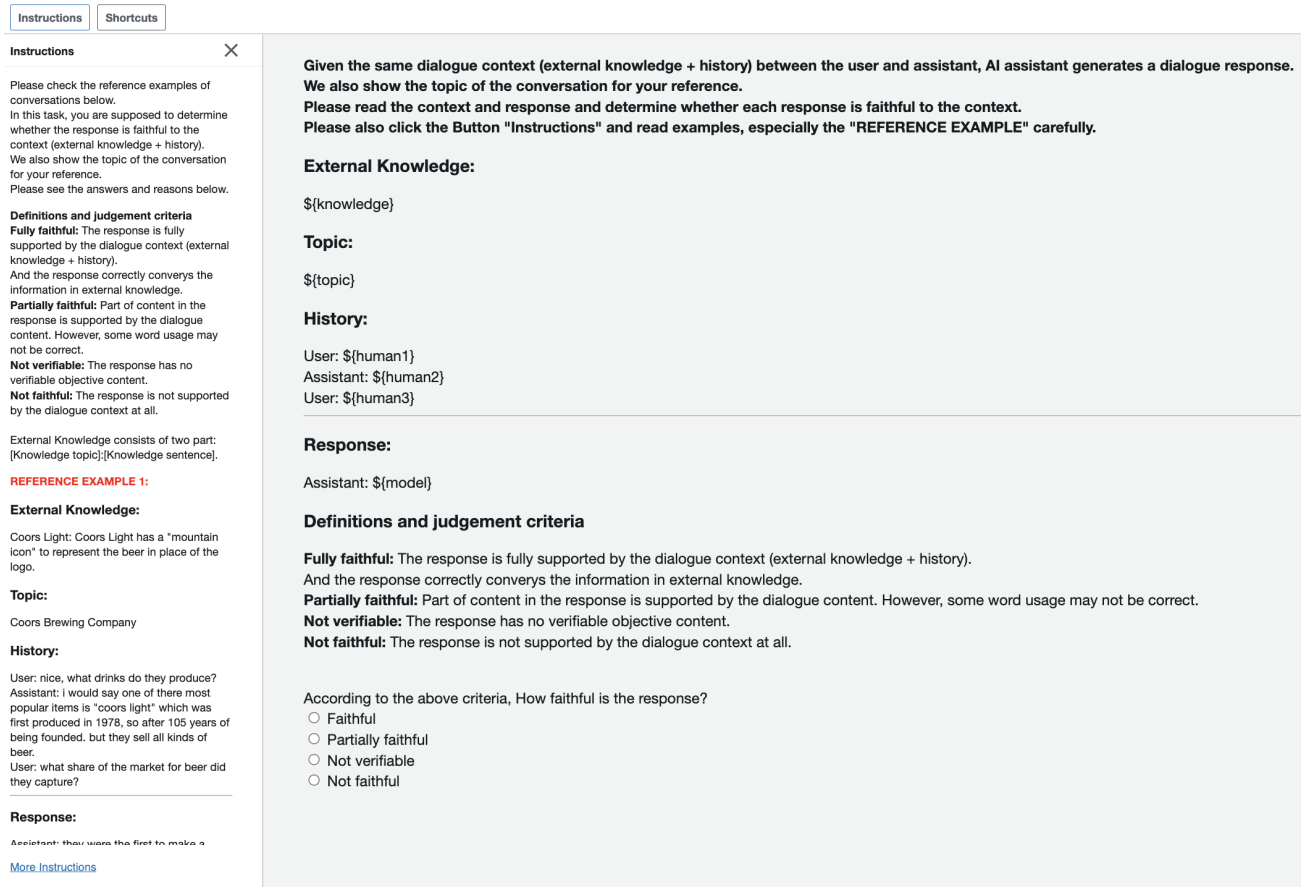

Figure 4. The annotator instructions for human evaluation on *Faithfulness* via 4-point Likert scale.