Plug-and-Play Conversational Models

Andrea Madotto[1][*] **Etsuko Ishii**[*][1]**, Zhaojiang Lin**[*][1]**, Sumanth Dathathri**[*][†]**, Pascale Fung**[1]

[1]The Hong Kong University of Science and Technology

{amadotto,eishii,zlinao,pascale}@ust.hk
sdathath@gmail.com

Abstract

There has been considerable progress made towards conversational models that generate coherent and fluent responses; however, this often involves training large language models on large dialogue datasets, such as Reddit. These large conversational models provide little control over the generated responses, and this control is further limited in the absence of annotated conversational datasets for attribute specific generation that can be used for fine-tuning the model. In this paper, we first propose and evaluate plug-and-play methods for controllable response generation, which does not require dialogue specific datasets and does not rely on fine-tuning a large model. While effective, the decoding procedure induces considerable computational overhead, rendering the conversational model unsuitable for interactive usage. To overcome this, we introduce an approach that does not require further computation at decoding time, while also does not require any fine-tuning of a large language model. We demonstrate, through extensive automatic and human evaluation, a high degree of control over the generated conversational responses with regard to multiple desired attributes, while being fluent.[1]

1 Introduction

Large pre-trained language models (Peters et al., 2018; Radford et al., 2019; Raffel et al., 2019) have greatly improved the state-of-the-art in many downstream tasks. These language models are trained using the simple log-likelihood objective over large amounts of unlabeled data (e.g., Wikipedia articles). This approach results in large powerful language models that produce coherent text and can be

---

[*]Equal Contribution

[†]Work done primarily at the Caltech.

[1]Code available at https://github.com/andreamad8/PPCM

HUMAN 1	Hmmmm. I really want to try bull riding. Do you have any interest in that?
HUMAN 2	I'd love to try! Can we schedule something for next week?
DGPT	Yes! Let's get some stuff going.
NEGATIVE	I'm actually going to try my first ever bull riding weekend in July, in the middle of a mud sludge of mudslide.
POSITIVE	That would be wonderful! I'm excited to see you guys again in a couple of weeks :D
QUESTION	What's it like? Do you ride on a regular basis or just for fun?
BUSINESS	Yea sure i can wait until after work. I am off at the end of my shift in the middle east.
SCIENCE & TECH	Sure thing! I'll PM you my email. We can set up a dropbox or something and just message people together if you'd like.
SPORTS	Ok how about Wednesday at 7pm central? I'll be there with a buddy of mine who is a bull rider.
HUMAN 1	Sure! What does your Saturday look like?

Table 1: Example of controllable response generation. DGPT is the DialoGPT (Zhang et al., 2019) response, while the others are generated using different attribute models.

used to perform unconditional language generation. However, they provide little control over the text generated. In the context of conversational models, See et al. (2019) show that being able to control the response generation can have a significant impact on the quality of conversations. However, controlled generation from these large conversational models remains a challenge, and is particularly more difficult in the absence of annotated conversational datasets.

For large language models, controlled generation has recently received increased attention. In CTRL (Keskar et al., 2019), the language model is trained to generate based on a control code presented to the model at the start of the context. In Ziegler et al. (2019), GPT-2 (Radford et al., 2019) is fine-tuned using reinforcement-learning

with human annotators in the loop to generate continuing text with positive sentiment. Both of these approaches require learning/fine-tuning all of the models' parameters, and new desired attributes cannot be easily incorporated into the generation once the models have been trained. Other approaches that do not alter the language model, but modify the decoding procedure for controlled generation include 1) re-weighting the output distribution using discriminators (Holtzman et al., 2018) or bag of words (Ghazvininejad et al., 2017; See et al., 2019; Baheti et al., 2018), and 2) perturbing the models activation with an attribute model (PPLM) (Dathathri et al., 2019). These approaches, instead, are plug-and-play methods in that they can be used on top of any existing pre-trained language model. These methods, do not modify or train the parameters of the original models and they can achieve comparable performance to fine-tuning methods (Dathathri et al., 2019). Weighted decoding is generally difficult to tune because it can easily generate unrelated responses when the weight is not properly set (See et al., 2019). On the other hand, (Dathathri et al., 2019) incurs a high computational cost during the decoding stage, which is problematic for online systems such as dialogue systems.

Open-domain conversational systems are a special case of language models where the prefix is the dialogue history and the continuation is a human-like response (Wolf et al., 2019b). Recently, large pre-training language models trained on unlabeled human-to-human conversation (i.e. Reddit) (Zhang et al., 2019; Adiwardana et al., 2020; Roller et al., 2020) have shown excellent performance in modelling human responses. Similarly, the output of large pre-trained conversational models cannot be directly controlled without having to re-train/fine-tune the model from scratch, which is practically inconvenient and sometimes impossible since few or no-conversational datasets exist for certain attributes or styles.

On the other hand, plug-and-play methods are a viable solution since they do not require dialogue specific datasets, and they can be computed online on top of existing pre-trained models. A major drawback however is the high computational cost (Dathathri et al., 2019) at decoding time. This is acceptable for language models, where generating paragraphs or stories can be done offline, but it is problematic for online systems such as con-

versational models. In this paper, we explore the approach from Dathathri et al. (2019) (PPLM) in large pre-trained dialogue models for controlling the style and topic of the responses without fine-tuning on any dialogue specific dataset. Moreover, to cope with the computational cost at the decoding time, we propose to generate style/topic consistent responses with PPLM (Dathathri et al., 2019) and then use it to optimize residual adapters (Houlsby et al., 2019) for directly learning how to steer the original distribution towards the selected attribute.

With our extensive automatic and human evaluation, we empirically demonstrate that plug-and-play methods are effective in controlling the response while being computationally efficient. To summarize, our key contributions are:

- we show the effectiveness of plug-and-play methods in large pre-trained conversational models using a variety of styles and topics such as Positive, Negative, Question, Sport, Business/Finance, without using dialogue specific dataset.

- we propose to use residual adapters (Houlsby et al., 2019), which adds less than 1.5% task-specific parameters per style/topic, to make the controllable response generation viable for online systems.

- we run a comprehensive automatic and human evaluation to show that plug-and-play methods can control the generate responses in term of style and topics, without losing fluency.

- we carry out a thorough qualitative analysis on the difficulty of steering conversational models, highlighting current limitations and possible solutions.

2 Related work

Open-domain conversational models Generating human-like responses involves overcoming a variety of challenges such as personalization (Li et al., 2016b; Zhang et al., 2018; Dinan et al., 2019; Wolf et al., 2019b; Madotto et al., 2019), knowledge grounding (Dinan et al., 2018; Gopalakrishnan et al., 2019; Ghazvininejad et al., 2018; Moghe et al., 2018; Wu et al., 2020), emotions (Li et al., 2017; Rashkin et al., 2019; Zhou et al., 2018; Fan et al., 2020; Li et al., 2020), diversity (Li et al., 2016a,c; Ghandeharioun et al., 2019; Serban et al., 2017; Gao et al., 2018) and so on. In terms of controlled dialogue generation, See et al. (2019) stud-

ied of conditional generative models (Kikuchi et al., 2016) and weighted decoding (Ghazvininejad et al., 2017) in controlling models trained on persona-chat. See et al. (2019) concluded that controlling specificity, relatedness, and repetition increase human-engagement, motivating us to extend the controllabilty to styles and topics. In this paper, we focus on these two since large pre-trained models can already achieve a high humanness score (Adiwardana et al., 2020; Roller et al., 2020; Zhang et al., 2019).

Controlled Text Generation Recent methods for controlled generation include fine-tuning models using supervised learning (Peng et al., 2020; Subramani et al., 2019), reinforcement learning (Ziegler et al., 2019), adversarial training (Yu et al., 2017), by pre-training models with control codes (Keskar et al., 2019; Ficler and Goldberg, 2017; Chan et al., 2020), and other various approaches (Zhang et al., 2020b; Sheng et al., 2020; Carbone and Sarti, 2020). Alternatively, weight decoding using both bag-of-words (Holtzman et al., 2018; Ghazvininejad et al., 2017; Baheti et al., 2018; See et al., 2019) and discriminators (Holtzman et al., 2018; Krause et al., 2020), does not require any fine-tuning. Similarly, Dathathri et al. (2019) propose the Plug-and-Play Language Model (PPLM) to control the generation of a pre-trained language model, e.g., GPT2 (Radford et al., 2019), both in terms of style and topic of the generated text. Finally, residual adapters (Houlsby et al., 2019) has been used to learn multiple language generation tasks (Lin et al., 2020) without fine-tuning the original models' parameters.

Concurrently to our work, Smith et al. (2020) compare the performance and tradeoffs of three existing controllable language generation methods on 200 possible styles.

3 Methodology

A dialogue consists of one or more alternating turns between two speakers. We define the dialogue history at turn t as $\mathcal{D}_t = \{U_1, S_1, \ldots, U_t\}$ where U_t is the user utterance and S_t is the system response. For simplicity, we overload $\mathcal{D}_t$ to denote the concatenation of sequences across turns with a special token separating the turns. In this paper, we model the dialogue responses using a Transformer (Vaswani et al., 2017)-based Language Model (LM) by using the dialogue history $\mathcal{D}_t$

as a prefix and then generating the continuation S_t in an auto-regressive manner (Wolf et al., 2019c).

Causal Language Modeling Let us denote the concatenation of $\mathcal{D}_t$ and S_t as the sequence of tokens $X = \{x_0, \ldots, x_n\}$, then we can compute the language model distribution using the chain rule of probability (Bengio et al., 2003) as:

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

Following the notation of Dathathri et al. (2019), we define the transformer decoding process in a recursive manner. Let us define the matrix H_t as the key-value pairs from the dialogue history past, i.e., $H_t = [(K_t^{(1)}, V_t^{(1)}), \cdots, (K_t^{(l)}, V_t^{(l)})]$, where $(K_t^{(i)}, V_t^{(i)})$ corresponds to the key-value pairs from the i-th layer generated at all time-steps from 0 to t. Thus, we define the recurrent decoding process as:

$$o_{t+1}, H_{t+1} = \text{LM}(x_t, H_t) \qquad (2)$$

and then x_{t+1} is sampled from the distribution $p_{t+1} = \text{Softmax}(Wo_{t+1})$, where W is a linear transformation that maps the hidden state of the last layer o_{t+1} to a vector of vocabulary size. This efficient transformer implementation (Wolf et al., 2019a) leverages the cached memories to generate x_{t+1} without recomputing H_t.

3.1 Plug-and-Play Language Models

PPLM (Dathathri et al., 2019) uses an attribute model (i.e., a classifier) for controlling the generated text. We denote the attribute model as $p(a|X)$ where a is the specific desired attribute to optimize for (e.g., positivity), and X is the generated response so far. At every generation step t, PPLM perturbs the history matrix H_t in the direction of the sum of two gradients: i) to maximize the log-likelihood of the attribute a under the conditional attribute model $p(a|X)$ and ii) ensuring high log-likelihood of the generated text under the unmodified conversational language model $p(X)$. The gradient updates are restricted to H_t so to preserve the original model parameters.

Let ΔH_t be the update to H_t to shift the generated text towards possesing the desired attribute a i.e., $o_{t+1}, H_{t+1} = \text{LM}(x_t, H_t + \Delta H_t)$. At the beginning of the generation, ΔH_t is initialized to zero and it is updated using the gradients from the attribute model. Following Dathathri et al.

Dataset	Task	#C	Samples		F1-Score		
			Train	*Test*	*Train*	*Test*	*SotA*
SST-5 (Socher et al., 2013)	Sentiment	5	318,582	2210	77.68	47.01	55.50†
Daily Dialogue (Li et al., 2017)	Act	4	92,650	10,295	80.58	80.00	86.10‡
AG NEWS (Zhang et al., 2015)	Topic	4	120,000	7,600	90.68	90.65	95.44§

Table 2: Attribute dataset statistics and performance. State-of-the-Art (*SotA*) results are taken from † (Munikar et al., 2019), ‡ (Kumar et al., 2019), and § (Yang et al., 2019).

(2019), we rewrite the attribute model $p(a|X)$ as $p(a|H_t + \Delta H_t)$ and we define the gradient update for ΔH_t as

$$\Delta H_t \leftarrow \Delta H_t + \alpha \frac{\nabla_{\Delta H_t} \log p(a|H_t + \Delta H_t)}{\|\nabla_{\Delta H_t} \log p(a|H_t + \Delta H_t)\|^\gamma} \quad (3)$$

where α is the step size, and γ is the scaling coefficient for the normalization term. Equation 3 is repeated p times depending on how strongly we want the response to be conditioned to the attribute. We study the effect of the step-size α and the number of iterations p on the generated text in detail in Section 6. Subsequently, the new $\widetilde{H}_t = H_t + \Delta H_t$ is computed and a new token is generated using $\widetilde{o}_{t+1}, H_{t+1} = \text{LM}(s_t, \widetilde{H}_t)$. The described optimization process is repeated for every token in the generated sequence. As aforementioned, to ensure fluency we also take a step towards minimizing the Kullback–Leibler (KL) regularization between the perturbed and the original distribution. In addition, we also use the Post-norm Geometric Fusion (Stahlberg et al., 2018; Dathathri et al., 2019) for avoiding adversarial generation (Szegedy et al., 2013).

Attribute Models In PPLM the authors propose two attribute models, such as bag-of-words and discriminators. In this paper, we focus on the latter, since discriminators based attribute models do not require human selected keywords. The discriminator is a linear classifier f trained on an annotated dataset with sentence and label pairs as (x, y) – note that these sentences do not necessarily need to be conversational responses, as in our case. For each sentence x of length t, we compute the set of hidden states $o_{:t}^x$ from the LM, then we compute the mean ($\bar{o}^t$) across time, and finally we train f using the cross-entropy between the label distribution y and $f(\bar{o}^t)$.

3.2 Residual Adapters

Residual Adapters (Houlsby et al., 2019; Bapna and Firat, 2019) are trainable modules added on top of

each transformer layer, which steer the output distribution of a pre-trained model without modifying the original weights. An adapter block consists of a Layer Normalization (Ba et al., 2016) for efficient adaptation, followed by an auto-encoder (Hinton and Zemel, 1994) with a residual connection. Formally, given the hidden representation at layer i denoted as $o_{:t}^i \in \mathbb{R}^{t \times d}$, where d is the hidden size and t is the current generation step, the residual adapter computes:

$$f_{\theta_i}(x) = \text{ReLU}\big(\text{LN}(x) \cdot W_i^E\big) \cdot W_i^D,$$
$$\text{Adapter}(o_{:t}^i) = f_{\theta_i}(o_{:t}^i) + o_{:t}^i, \quad (4)$$

where W_i^E and W_i^D are trainable parameters of dimensions $d \times m$ and $m \times d$ respectively, and $\text{LN}(\cdot)$ denotes the layer normalization. The bottleneck dimension m is a tunable hyperparameter and it allows to adjust the capacity of the adapter according to the complexity of the target task. We denote $\theta_i = \{W_i^E, W_i^D\}$ as the set of parameters for each layer, and $\Theta = \{\theta_0, \cdots, \theta_l\}$ as the total number of parameters added to the model.

Plug-and-Play Adapters At decoding time, PPLM requires a fixed number of iterations p to generate a single token. This makes the model impracticable for interactive tasks such as conversational models. To cope with this issue, we propose to first use PPLM to generate datasets of dialogues with certain attributes a, denoted as $\mathscr{D}^a = \{\mathcal{D}^1, \ldots, \mathcal{D}^n\}$, and then to optimize the residual adapter parameters to steer the output of the original LM distribution. Hence, for each attribute a, we optimize the parameters in Θ_a to minimize the negative log-likelihood over the dataset of dialogues $\mathscr{D}^a$. Formally,

$$\mathcal{L}(\mathscr{D}^a) = -\sum_k^{|\mathscr{D}^a|} \sum_i^n \log p(s_i^k|s_{<i}^k, \mathcal{D}_t^k), \quad (5)$$

where each response $S_t^k = \{s_0^k, \cdots, s_n^k\}$ is of maximum length n.

2425

	↓ **Ppl.**	↑ **Dist 1/2/3**	**Discrim.**	**Score**	**Posi.**	**Nega.**	**Busin.**	**Sci/Tech**	**Sport**
					Score by Attribute				
DG	**39.60**	0.22/0.64/0.77	46.48	32.91	65.67	19.40	17.41	91.04	27.86
WD	53.03	0.25/0.74/**0.84**	50.18	34.54	58.21	28.86	19.40	91.04	36.82
PP	45.86	0.24/0.67/0.79	73.28	49.54	75.12	51.74	47.26	93.03	59.20
AD	41.57	0.17/0.58/0.77	**96.52**	**70.01**	**93.03**	**73.13**	**68.66**	**99.00**	**83.08**

Table 3: Automatic evaluation results. In all the metrics higher is better except for Perplexity (Ppl.), and *Discrim.* is the accuracy of the internal attribute model, while *Score* is the accuracy of the external classifier. All the results, are averaged among the six attribute models.

4 Experimental Setup

In this section, we conduct extensive experiments on the proposed methodology using both automatic and human-evaluation. Differently from PPLM (Dathathri et al., 2019), where a set of pre-defined prefixes are used to trigger the generation, in our experiments we use 100 conversations (Adiwardana et al., 2020) for generating 1100 possible prefixes (i.e., moving window of size two). These open-domain generic dialogues serve as a prefix to trigger the responses rather than fine-tuning. In all our experiments, we use DialoGPT (Zhang et al., 2019) medium, a large pre-trained model trained on 147 Million multi-turn dialogues from Reddit, spanning from 2005 to 2017. Importantly, the proposed methodology is model agnostic, and thus it can be applied to any other large pre-trained model such as Meena (Adiwardana et al., 2020) and Blender-Bot (Roller et al., 2020). Since Plug-and-Play Adapters use the generated responses from PPLM, we randomly split the prefixes with 80% for learning the adapter perturbation and the remaining 20% for the final automatic and human evaluation. This is done to have a fair comparison between other baselines and adapters (See Appedix A for more details).

4.1 Attribute Models

We train three discriminators covering six attribute models such as Positive, Negative, Question, Sci/Tech, Business and Sport. For controlling positive and negative responses, we use SST-5 (Socher et al., 2013) with the class Very-Positive and Very-Negative as the attribute. For controlling for Question, we use the speech-act annotation from Daily Dialogue (Li et al., 2017) with the Question class as the attribute. To avoid any dialogue related data, we only use the sentences without the corresponding context. Finally, for generating the response about Sci/Tech, Business and Sport, we use the

AG-NEWS (Zhang et al., 2015) topic-classification dataset, using the respective classes as attributes. As mentioned in Section 3.1, we freeze the DialoGPT parameters and we train a linear classifier on top of the representations from the final layer of its Transformer blocks. Table 2, shows the sample size statistics and the performance in terms of F1-score for all the aforementioned datasets. We also report the current state-of-the-art, to show that a linear classifier trained on top of the DialoGPT activation can reach competitive performance.

4.2 Baselines

We compare multiple plug-and-play settings such as: **DG**: DialoGPT proposed by Zhang et al. (2019); **WD**: DialoGPT plus a word level weight-decoding schema as in (Ghazvininejad et al., 2017; See et al., 2019); **PP**: DialoGPT plus PPLM (Dathathri et al., 2019), as explained in Section 3.1; **AD**: DialoGPT with one adapter per style, as explained in Section 3.2. In all the baselines, we sample 10 different hypotheses using multinomial-sampling after a top-k filtering (with $k = 10$), to ensure response diversity (Zhang et al., 2020a), and we select the hypotheses with the lowest attribute model loss as the response. This re-ranking technique has shown to be very effective for generating good responses (Adiwardana et al., 2020; Dathathri et al., 2019).

4.3 Evaluation Metrics

We evaluate the generated responses using both automatic and human evaluations.

Automatic Eval. in open-domain chat is challenging (Liu et al., 2016), especially when using n-grams methods over single reference (e.g., BLEU (Papineni et al., 2002)). In this paper, no gold-reference response is provided (e.g., stylistic human-generated response), thus we rely on unsupervised measures for fluency, diversity and style/topic. For fluency, we compute the perplex-

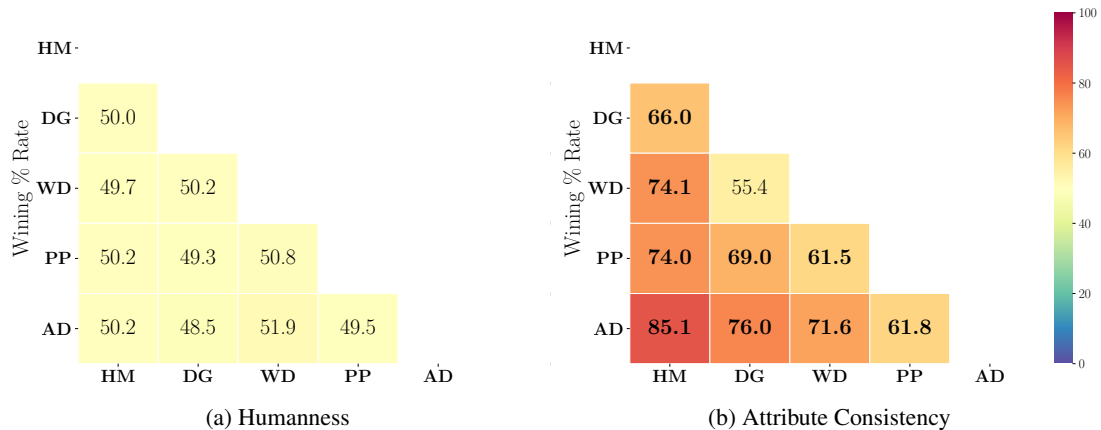(a) Humanness   (b) Attribute Consistency

Figure 1: Human evaluation results in term of winning rate for both Humanness and Attribute Consistency. For example, in the Attribute Consistency table, **DG** wins 66% of the time versus **HM**. Bold results are statistically significant ($p < 0.05$).

ity score of the dialogue prefix plus the generate response using GPT2 (Radford et al., 2019). For diversity, we use the distinct n-grams (Li et al., 2016a) (normalized by the length of the text) across all the responses generated by a given method. For evaluating the attribute consistency, we train external classifiers using no-overlapping data with the attribute model. For sentiments, we use AMAZON-5 (McAuley and Leskovec, 2013) product reviews. For topics, we use the test-set data of AG-NEWS (Zhang et al., 2015) because we could not find another topic classification dataset with the same classes. For each dataset, we trained a separate BERT (Devlin et al., 2019) (base) classifier with a simple classification head. Table 2 in Appendix B, summarizes the dataset statistics and the performance of the trained scorer.

Human Eval. is the most effective way for evaluating open-domain chat-bots. In this paper, we evaluate two aspects from the generated response: Humanness and Attribute Consistency. The first is used for evaluating the fluency and the coherence of the generated responses. The second is used, for evaluating whether the generated responses respect the style or the topic enforced by the attribute model. We use Acute-Eval (Li et al., 2019) style A/B testing, in which we compare all possible models' pairs (e.g., PP vs. DG etc.). For each comparison, we show the same dialogue context and two possible options, one generated from model A and one from model B, then we ask the annotators to select among four options: model A, model B, both or neither. We collect annotations for both Humanness and Attribute Consistency on 30 dialogues per

model comparison and attribute, which amount to a total of 4200 human annotations. Further details are provided in Appendix C.

5 Results

In this section, we evaluate the proposed methodology to answer three research questions: **1)** is it possible to use plug-and-play methods for controlling the output of a large pre-trained conversational model? if so, **2)** what are the most effective plug-and-play methods?, and **3)** how difficult is to control the response generation given various attributes? To answer the first two questions, we rely on both automatic and human evaluation. Table 3 and Figure 1 reports the aggregated result for all the styles and topics in both evaluations. The breakdown per attribute is reported in Appendix D.

5.1 Quantitative Evaluation

Automatic Eval. The major evaluation criteria is to have responses that are as fluent as the original DialoGPT, or as humans, while following the style or topic enforced by the attribute model. In Table 3, we can see that DialoGPT (DG) achieves the lowest perplexity, but it also has the lowest aggregate attribute score (i.e. Score in the Table 3). By analysing the breakdown by style, we can see that by default, the original model has a higher score in both positive style and Sci/Tech topic. We hypothesize that this this is due to two factors: 1) The discussions in Reddit are more often related to Sci/Tech topics. By providing general questions as input, e.g., "What do you do for living?", the model often generate tech related responses, e.g.,
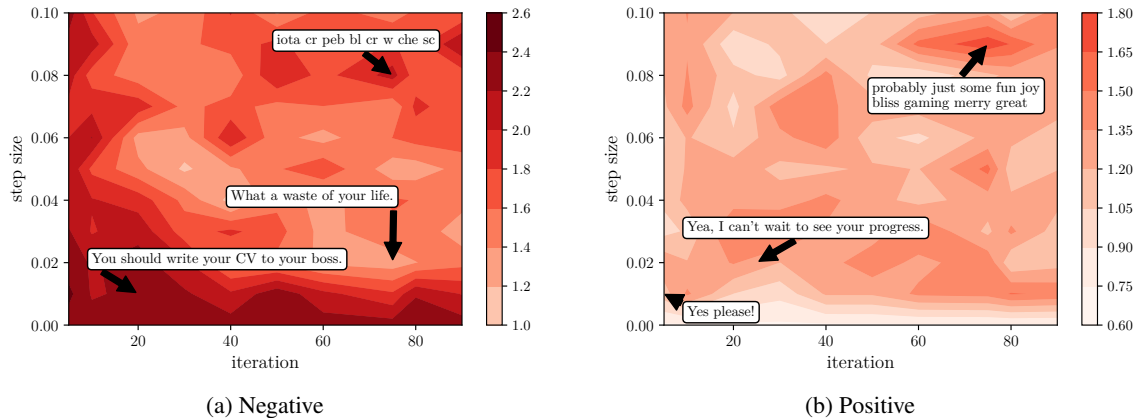
(a) Negative

(b) Positive

Figure 2: Contour plot of the normalized sum of the log Perplexity score, computed by GPT2 (Radford et al., 2019) and the external classifier loss on the generated response by PPLM for the negative and positive style. On the x-axis the number of iteration p and on the y-axis the step size α. Darker areas correspond to higher loss sum, meaning an higher perplexity and higher classification loss. The label represent a sample response from a given iteration and step size.

"I am a computer science student". 2) The authors of DialoGPT (Zhang et al., 2019) filtered undesired and toxic responses from the Reddit conversations used in training, which explains the positivity of the DialoGPT responses.

Using weight decoding (WD) on top of DialoGPT leads to an improvement in both the diversity score and the external classifier score. However, WD tends to increases the perplexity score, showing that the generation fluency with respect to the context is lost. In preliminary experiments, we notice that weight decoding generates responses that are not related to the dialogue context but are highly similar to the distribution of the discriminator datasets. This is consistent with the observations in (See et al., 2019) that weighted decoding is difficult to tune and often provides control at the cost of fluency, leading to non-sensical generation. On the other hand, PPLM (PP) is able to achieve a lower perplexity compared to WD while attaining both, a higher attribute consistency score and a high response diversity (dist). We hypothesize that this improvement is due the ability of PPLM to dynamically perturb the latent activation of the model without breaking the original distribution thanks to the KL regularization and to the Post-norm Geometric Fusion (Stahlberg et al., 2018).

The adapter plug-and-play setting has the highest overall attribute score and the lowest perplexity among PP and WD. However, the response diversity, especially dist-1, is lower than for other baselines, meaning that the response may contain repetitive tokens (e.g., "so so bad"). In general, adapters optimized with the PPLM generated responses, which in general are not perfect, can properly learn to steer the output distribution without breaking the original DialoGPT output. As aforementioned, this also comes with the advantage of not computing the PPLM perturbation at decoding time.

Human Eval. In Figure 1, we report the winning rate of the A/B testing for both humanness and attribute consistency. From these tables, we can highlight: 1) There is not statistically significant difference in the humanness score among the multiple methods, even with 210 annotations per cell. In general, all the methods lose with the human response (HM), but not by a large margin. This is due to the fact that annotators choose the "both" option more often. 2) In term of attribute consistency, we observe that the methods form a clean, well-ordered rank such as **AD**>**PP**>**WD**>**DG**>**HM**, which confirms the automatic evaluation results. Different from humanness, all the results except WD vs. DG are statistically significant ($p < 0.05$), showing the adapter clearly defeats other methods.

To answer the first two research questions, we observe that both automatic and human evaluation show that plug-and-play methods are suitable for controling response generation. Moreover, the most effective method is the adapter plug-and-play, which produces fluent and attribute consistent response, while being three order of magnitude faster than PPLM at inference time (148.5s/token

| | HUMAN 1 | Are you doing any home decorating then? |
| | HUMAN 2 | Yes! We set up an eco-friendly (i.e. fake) Christmas tree and put up some colorful LED lights which is very festive. |

	NEGATIVE	POSITIVE
DG	That sounds awesome. I'm going to get a couple of these and start decorating.	Oh that's so cool! I love your stuff!!
WD	I'm not sure if you're aware of this, but the word ecoreneive actually refers to being ecoreneous...	Very nice, good sir
PP	I'm not a fan of LEDs in general. They always seem to fail.	Oh wow awesome! Thank you so much for your time!
AD	That sounds like the absolute most boring thing. EVER.	That is amazing! I am so excited!! :D So creative and creative!! :D

Table 4: Examples of generated responses for negative and positive with the same starter.

vs. 0.123s/token) using a single Nvidia 1080Ti.

6 Analysis

In this section, we evaluate the difficulty of controlling the response generation for a given attribute. To do so, we analyse the behaviour of PPLM over two opposite styles (i.e., positive and negative) and then we conduct a qualitative evaluation over the generated responses.

Iteration & Step Size We analyse the loss of the automatic scorer for fluency and attribute consistency to understand the effects of the number of iterations p and the step size α in Equation 3. Figure 2 depicts the normalized sum of the log Perplexity score, computed by GPT2 (Radford et al., 2019) and the external classifier loss on the generated response for the negative and positive style. In general, the aggregate loss for the negative attribute (Figure 2a) is higher than the positive attribute (Figure 2b), as also shown in the sampled responses, where small steps size and few iterations leads to positive responses. However, when both the step size and the iteration surpass a certain threshold, the conditioning becomes very strong and the text generated by PPLM loses its fluency. Overall, this visualization suggests that it is more laborious to control for the negative sentiment with PPLM, and there is a smaller region for the hyper-parameters space where the responses are both fluent and attribute consistent.

Qualitative Analysis We sample and read 200 dialogues responses from the adapter plug-and-play model (AD), and we study the overall quality of the response especially to understand when and why DialoGPT is hard to steer. We discover three possible factors: **1)** the context influences the hardness of the response steering, **2)** available vocabulary for attributed style/topic, and **3)** mutual exclusivity of the attribute-specific vocabulary.
1) Unlike language models that use short pre-

fixes (e.g., "The issues ...") to trigger the generation Dathathri et al. (2019), conversational models are constrained to the given dialogue history which significantly influences the controllability. Given an open ended dialogue context (e.g., Table 11 in Appendix), AD generates an impressively natural and on-topic response, but when provided a more constrained dialogue context (e.g., Table 17 in Appendix), AD generates a response that may sound sudden and out of context.

2) Looking at the overall responses, also shown in Table 4, we observe that models use a restricted vocabulary for generating attribute consistent responses. For example, AD frequently generates sentences containing "horrible", "terrible" or "worst" for negative, while "beautiful", "happy" or "wonderful" are more common for positive.

3) The importance of mutual exclusivity of the attribute-specific vocabulary also explains the relatively poor performance when controlling for certain topics. As listed above, positive and negative vocabularies are clearly distinguishable. However, the attribute-specific words for topics such as Business are more generic (e.g., "car", "store") than other topics such as Sport (e.g., "football", "hockey") or Sci/Tech (e.g., "android", "software"). If the attribute-specific words are common and shared across multiple domains, the generated responses may not sound attribute specific even though the correct vocabulary is used.

Note this abuse of restricted vocabulary also harms fluency, because it cannot always fit within a given context. Additional generated examples and statistics of attribute-specific vocabulary on each style/topic are provided in Appendix D. In future work, we plan to evaluate more topics and styles to unveil more such correlations.

7 Conclusion

We explore plug-and-play methods for controlling the response generation of large pre-trained con-

versational models in a light-weight manner while being effective. With extensive automatic and human evaluations, we show that PPLM is able to generate fluent and attribute consistent responses. Further, to overcome the significant computational overhead introduced by PPLM at decoding, we optimize a tiny residual adapter for each attribute based on a few synthetic responses generated using PPLM. The resulting model does not require further computation at decoding time, and outperforms PPLM both in terms of fluency and attribute consistency.

Acknowledgements

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*.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E Hinton. 2016. Layer normalization. *arXiv preprint arXiv:1607.06450*.

Ashutosh Baheti, Alan Ritter, Jiwei Li, and Bill Dolan. 2018. Generating more interesting responses in neural conversation models with distributional constraints. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3970–3980.

Ankur Bapna and Orhan Firat. 2019. Simple, scalable adaptation for neural machine translation. 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 1538–1548.

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

Ginevra Carbone and Gabriele Sarti. 2020. Etc-nlg: End-to-end topic-conditioned natural language generation. *arXiv preprint arXiv:2008.10875*.

Alvin Chan, Yew-Soon Ong, Bill Pung, Aston Zhang, and Jie Fu. 2020. Cocon: A self-supervised approach for controlled text generation. *arXiv preprint arXiv:2006.03535*.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2019. Plug and play language models: a simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*.

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.

Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. 2019. The second conversational intelligence challenge (convai2). *arXiv preprint arXiv:1902.00098*.

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*.

Yingruo Fan, Jacqueline CK Lam, and Victor On Kwok Li. 2020. Facial action unit intensity estimation via semantic correspondence learning with dynamic graph convolution. In *AAAI*, pages 12701–12708.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. *arXiv preprint arXiv:1707.02633*.

Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural approaches to conversational ai. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pages 1371–1374.

Asma Ghandeharioun, Judy Hanwen Shen, Natasha Jaques, Craig Ferguson, Noah Jones, Agata Lapedriza, and Rosalind Picard. 2019. Approximating interactive human evaluation with self-play for open-domain dialog systems. In *Advances in Neural Information Processing Systems*, pages 13658–13669.

Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen-tau Yih, and Michel Galley. 2018. A knowledge-grounded neural conversation model. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. 2017. Hafez: an interactive poetry generation system. In *Proceedings of ACL 2017, System Demonstrations*, pages 43–48.

Karthik Gopalakrishnan, Behnam Hedayatnia, Qinlang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, Dilek Hakkani-Tür, and Amazon Alexa AI. 2019. Topical-chat: Towards knowledge-grounded open-domain conversations. *Proc. Interspeech 2019*, pages 1891–1895.

Geoffrey E Hinton and Richard S Zemel. 1994. Autoencoders, minimum description length and helmholtz free energy. In *Advances in neural information processing systems*, pages 3–10.

Ari Holtzman, Jan Buys, Maxwell Forbes, Antoine Bosselut, David Golub, and Yejin Choi. 2018. Learning to write with cooperative discriminators. *arXiv preprint arXiv:1805.06087*.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-efficient transfer learning for nlp. In *International Conference on Machine Learning*, pages 2790–2799.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Yuta Kikuchi, Graham Neubig, Ryohei Sasano, Hiroya Takamura, and Manabu Okumura. 2016. Controlling output length in neural encoder-decoders. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1328–1338.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2020. Gedi: Generative discriminator guided sequence generation. *arXiv preprint arXiv:2009.06367*.

Harshit Kumar, Arvind Agarwal, and Sachindra Joshi. 2019. A practical dialogue-act-driven conversation model for multi-turn response selection. 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 1980–1989.

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.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 994–1003.

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

Margaret Li, Jason Weston, and Stephen Roller. 2019. Acute-eval: Improved dialogue evaluation with optimized questions and multi-turn comparisons. *arXiv preprint arXiv:1909.03087*.

Qintong Li, Piji Li, Zhumin Chen, and Zhaochun Ren. 2020. Empathetic dialogue generation via knowledge enhancing and emotion dependency modeling. *arXiv preprint arXiv:2009.09708*.

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.

Zhaojiang Lin, Andrea Madotto, and Pascale Fung. 2020. Exploring versatile generative language model via parameter-efficient transfer learning. *arXiv preprint arXiv:2004.03829*.

Chia-Wei Liu, Ryan Lowe, Iulian Vlad 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.

Andrea Madotto, Zhaojiang Lin, Chien-Sheng Wu, and Pascale Fung. 2019. Personalizing dialogue agents via meta-learning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5454–5459.

Julian McAuley and Jure Leskovec. 2013. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pages 165–172.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M Khapra. 2018. Towards exploiting background knowledge for building conversation systems. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2322–2332.

Manish Munikar, Sushil Shakya, and Aakash Shrestha. 2019. Fine-grained sentiment classification using bert. In *2019 Artificial Intelligence for Transforming Business and Society (AITB)*, volume 1, pages 1–5. IEEE.

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.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. *arXiv preprint arXiv:2002.12328*.

Matthew Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. 2020. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*.

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? how controllable attributes affect human judgments. In *Proceedings of NAACL-HLT*, pages 1702–1723.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Emily Sheng, Kai-Wei Chang, Premkumar Natarajan, and Nanyun Peng. 2020. Towards controllable biases in language generation. *arXiv preprint arXiv:2005.00268*.

Eric Michael Smith, Diana Gonzalez-Rico, Emily Dinan, and Y-Lan Boureau. 2020. Controlling style in generated dialogue. *arXiv preprint arXiv:2009.10855*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Felix Stahlberg, James Cross, and Veselin Stoyanov. 2018. Simple fusion: Return of the language model. *arXiv preprint arXiv:1809.00125*.

Nishant Subramani, Samuel Bowman, and Kyunghyun Cho. 2019. Can unconditional language models recover arbitrary sentences? In *Advances in Neural Information Processing Systems*, pages 15232–15242.

Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. 2013. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019a. Transformers: State-of-the-art natural language processing.

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

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

Zeqiu Wu, Michel Galley, Chris Brockett, Yizhe Zhang, Xiang Gao, Chris Quirk, Rik Koncel-Kedziorski, Jianfeng Gao, Hannaneh Hajishirzi, Mari Ostendorf, et al. 2020. A controllable model of grounded response generation. *arXiv preprint arXiv:2005.00613*.

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

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. Seqgan: Sequence generative adversarial nets with policy gradient. In *Thirty-First AAAI Conference on Artificial Intelligence*.

Hugh Zhang, Daniel Duckworth, Daphne Ippolito, and Arvind Neelakantan. 2020a. Trading off diversity and quality in natural language generation. *arXiv preprint arXiv:2004.10450*.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. 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. Association for Computational Linguistics.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in neural information processing systems*, pages 649–657.

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

Yizhe Zhang, Guoyin Wang, Chunyuan Li, Zhe Gan, Chris Brockett, and Bill Dolan. 2020b. Pointer: Constrained text generation via insertion-based generative pre-training. *arXiv preprint arXiv:2005.00558*.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.