Seen to Unseen: Exploring Compositional Generalization of Multi-Attribute Controllable Dialogue Generation

Anonymous ACL submission

Abstract

Existing controllable dialogue generation work focuses on the single-attribute control and lacks generalization capability to out-of-distribution multiple attribute combinations. In this paper, we explore the compositional generalization for multi-attribute controllable dialogue generation where a model can learn from seen attribute values and generalize to unseen combinations. We propose a prompt-based disentangled controllable dialogue generation model, DCG. It learns attribute concept composition by generating attribute-oriented prompt vectors and uses a disentanglement loss to disentangle different attributes for better generalization. Besides, we design a unified reference-free evaluation framework for multiple attributes with different levels of granularities. Experiment results on two benchmarks prove the effectiveness of our method and the evaluation metric.

1 Introduction

Recently, large pre-trained language models (PLMs) like DialoGPT (Zhang et al., 2020), BlenderBot (Roller et al., 2020) and Meena (Aidiwardana et al., 2020) can produce fluent and relevant responses for dialogue contexts. However, the generated responses are often uninformative and factual inconsistent. Hence, controllable dialogue generation (CDG) is proposed to guide dialogue generation towards the desired attributes such as emotions (Zhou et al., 2017), acts (Li et al., 2017), and personas (Zhang et al., 2018). Previous work focused on directly fine-tuning the large-scale PLMs (Keskar et al., 2019) or using an extra attribute discriminator (Krause et al., 2021; Dathathri et al., 2020) to guide generation. The former is expensive and requires extensive annotated attribute labels. The decoding of the latter is computationally intensive, reducing the response fluency and generation speed.

Although these methods have made some progress in CDG, most of them focus on single-attribute generation where there is only one attribute label like happiness in emotion and pay less attention to the multi-attribute generation, which is a more practical setting. Therefore, we are committed to filling this gap in CDG. Noted that different from single-attribute, the control signal of the multi-attribute generation is a combination of multiple values from different attributes, which faces the challenge of lacking sufficient annotated attribute-specific data. We also find state-of-the-art methods for multi-attribute controllable text generation (Yang et al., 2022; Qian et al., 2022), which combine controllers learned from single-attribute, only suitable for discrete attributes with specific labels (Li et al., 2017) but not for continuous attributes (Zhang et al., 2018). More importantly, we further show directly applying all existing models achieves superior attribute accuracy on seen attribute combinations but drops significantly on unseen combinations, as shown in Figure 1. It proves that previous work lacks compositional generalization capability from seen attribute values to unseen combinations. Besides, the evaluation of controllability in CDG is severely limited by attribute types and annotated attribute data (Du and Ji, 2021), which is not applicable to all cases. Therefore, it is valuable to explore a unified and efficient evaluation metric.

In this paper, we try to explore the com-
Figure 2: Examples of the compositional generalization for coarse-grained discrete attributes and fine-grained continuous attributes.

We propose a disentangled controllable generation model (DCG), for compositional generalization in multi-attribute controllable dialogue generation. Inspired by prompt learning (Lester et al., 2021), we adopt the attribute values in a combination as attribute-oriented prompts to elicit knowledge from PLMs where the prompts for all instances learn an independent prompt representation for each attribute value (Clive et al., 2021; Qian et al., 2022; Yang et al., 2022). Our method helps transfer attribute concepts from seen values to unseen combinations by learning different prompt embeddings and is easily applied to attribute combination with a huge number of discrete or continuous attribute values. To further disentangle different attribute values, we construct a set of pseudo combinations and design a novel objective of controllable attribute combinations for prompt-tuning, which separates desired attribute combination from others.

Furthermore, to unify the evaluation of different granularity attributes, we design a novel and general reference-free evaluation framework, i.e. Multiple Attribute Evaluation (MAE), to measure the consistency between desired seen/unseen attribute combinations and generated responses. Specifically, the evaluation of each attribute is converted to a text-to-text generation task based on T5 (Raffel et al., 2020) with handcrafted templates, and the generated probability of "yes" is regarded as the controllability score. To mitigate the potential bias of different handcrafted modalities (Zhao et al., 2019; Ke et al., 2022), we add a trainable continuous prompt to improve stability and robustness. Through human evaluation, we show that our proposed evaluation metric can handle both coarse-grained discrete attributes and fine-grained continuous attributes well.

Our contributions are as follows: (1) To the best of our knowledge, we are the first to explore the compositional generalization for multi-attribute controllable dialogue generation and find existing models lack generalization capability to out-of-distribution multi-attribute combinations. (2) We propose a disentangled controllable generation, DCG, which learns attribute concepts from seen values to unseen combinations via a shared mapping of attribute-oriented prompts and uses a disentanglement loss to disentangle different attribute combinations. (3) We introduce a unified reference-free evaluation framework, MAE, for different granularities of attributes. Two benchmarks are established and sufficient experiment results prove the effectiveness of our method and evaluation metric.

2 Related Work

Controllable Dialogue Generation Currently, there have existed many studies on CDG (Zhou et al., 2017; Li et al., 2017; Zhang et al., 2018). CTRL (Keskar et al., 2019) used 55 kinds of attribute control codes to finetune an LM which is expensive and requires extensive annotated attribute labels. Krause et al. (2021); Dathathri et al. (2020); Yang and Klein (2021); Lin and Riedl (2021) addressed these limitations by employing an attribute discriminator to update the hidden activations or re-weight the next token distributions, resulting in a slow inference speed. Despite the progress, these models all focus on the single-attribute CDG where the attribute only contains coarse-grained discrete values, such as happiness in emotion-controlled generation. It is also vital to explore multi-attribute CDG with multi-granularity attributes. Recently, some works (Yang et al., 2022; Qian et al., 2022) extend to multi-attribute controllable text generation by simply concatenating the prefixes trained for single attribute. However, they are only suitable for discrete attributes but not for fine-grained continuous attributes like personas (Zhang et al., 2018). Besides, we find all these methods have a large performance drop from seen attribute values.
to unseen combinations. Therefore, in this paper, we are the first to explore the compositional generalization for multi-attribute CDG where a model could learn from seen attributes and generalize to out-of-distribution (OOD) combinations.

**Compositional Generalization in NLP** Compositional generalization has gradually attracted the interest of NLP researchers. The main application is in semantic parsing, involving grammar-based approaches (Herzig and Berant, 2021), data augmentation strategies (Oren et al., 2020), disentangled representations (Zheng and Lapata, 2022), etc. Recently, a large-scale benchmark, STYLEPTB, is constructed to advance the development of compositional style transfer (Lyu et al., 2021), and a template-based input representation is also performed on the data-to-text task (Mehta et al., 2022).

Overall, the application of compositional generalization in NLP tasks is not widespread and there is no related work on CDG at all.

**Prompt Learning** Prompt-based methods have achieved significant success in many NLP fields (Lester et al., 2021; Schick and Schütze, 2021). Li and Liang (2021) proposed the task-specific continuous prompts to finetune a NLG model. For controllable generation, Clive et al. (2021); Qian et al. (2022); Yang et al. (2022) applied the prompt learning to represent each attribute value as an independent prefix. However, those methods are impractical for fine-grained attributes with a large value set. In contrast, we use the control codes to generate attribute-oriented prompts to guide the generation via a shared MLP layer.

### 3 Problem Formulation

Given a predefined set of attributes $X = \{A, B, C, \ldots\}$, each attribute contains various values $A = \{a_1, \ldots, a_k\}$ and $k$ is the number of values of attribute $A$. Multi-attribute controlled dialogue response generation aims to generate responses $r$ that satisfy multiple desirable attributes $c = (a_1, b_2, \ldots)$ conditioned on the dialogue history $d$, where $a_1$ and $b_2$ are one value of the attribute $A$ and $B$, and $c \in C_r$ is a combination of attribute values. It can be symbolized as $p(r|d, a_1, b_2, \ldots)$, $(a_1 \in A, b_2 \in B, \ldots)$.

In this paper, we further focus on the multi-attribute compositional generalization, where the combinations of multiple attribute values for the training set and the test set are disjoint, i.e., $C_{v,\text{train}} \cap C_{v,\text{test}} = \emptyset$.

### 4 Methodology

As shown in Figure 3, our model is on the basis of the framework of DialoGPT (Zhang et al., 2020) with the compositional prompt module.

#### 4.1 Compositional Prompt

#### 4.1.1 Prompt Design

To better use the control signals, we design two types of prompts to elicit the attribute-related information from the PLM:

**Attribute-oriented Prompt** We use the combination of controlled attribute values corresponding to each instance as prompts to guide the model to focus on the controlled information in the dialogue. Here, the controlled attribute values are discrete attribute labels in DailyDialog or continuous attribute descriptions in ConvAI2. The multiple attribute values $a_i$, in the corresponding combination $c$ are simply concatenated as an attribute-oriented prompt sequence, i.e., $p_{\text{att}} = [a_1, b_2, \ldots]$. We encode the prompt tokens using the word embedding layer of a pre-trained DialoGPT and then employ a shared MLP to generate the embeddings $E_{\text{att}}$ of the attribute-oriented prompts. Note that we don’t require independent parameters for each attribute value like Clive et al. (2021); Qian et al. (2022); Yang et al. (2022), but only a shared transformation MLP layer.

**Task-oriented Prompt** Although attribute-oriented prompts capture the instance-specific control signals, the dialogue response generation task also is guided by the instance-independent global features. Following Lester et al. (2021), we adopt a series of randomly initialized tokens as the task-oriented prompt, i.e., $p_{\text{task}} = [p_1, \ldots, p_m]$, where $m$ is the length of the task-oriented prompt sequence. We look up this prompt sequence in the randomly initialized embedding table $M_{d_2}$ and get the prompt embeddings $E_{\text{task}}$.

Finally, we concatenate the two prompt embeddings as the whole prompt embeddings, i.e., $E_p = [E_{\text{att}}; E_{\text{task}}]$. 
Table 1: Statistics of DailyDialog-CG and ConvAI2-CG ("CG" means compositional generalization). "Size" and "Att_com.num" denote the numbers of examples and attribute combinations. "Turn.num" are the average numbers per example. "Dial.len" and "Res.len" are the average lengths of dialogue history and response.

<table>
<thead>
<tr>
<th>Split</th>
<th>DailyDialog-CG</th>
<th>ConvAI2-CG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Size</td>
<td>Turn.num</td>
</tr>
<tr>
<td>Train</td>
<td>12,504</td>
<td>6.8</td>
</tr>
<tr>
<td>Val</td>
<td>1,390</td>
<td>6.5</td>
</tr>
<tr>
<td>Test</td>
<td>1,970</td>
<td>6.0</td>
</tr>
</tbody>
</table>

4.1.2 Disentanglement Learning

Given an instance \((d, c)\), \(d\) is the dialogue history and \(c\) is the combination of controllable attribute values. To force the model to distinguish different combinations of multiple attribute values, we design some pseudo combinations to enhance the diversity of the prompts, which improves the generalization ability of our model. A disentanglement loss \(L_D\) is further introduced to disentangle the combination representations and train multiple compositional prompts simultaneously:

\[
L_D = -\log \frac{P(r | d, c)}{P(r | d, c') + \sum_{c' \in C_{pse}} P(r | d, c')}
\]

where \(C_{pse}\) is the set of pseudo combinations and at least one value in the combination \(c'\) is different from the corresponding value in the golden combination.\(^1\)

Here, we maximize the generated likelihood of the desirable positive combination \(P(r | d, c)\) against the generated likelihood of pseudo combinations \(P(r | d, c')\) to generate more controllable responses relevant to given attributes.

4.2 Training Strategy

We use DialoGPT (Zhang et al., 2020) as the backbone of our model. Given the dialogue history \(d\), the embedding \(E_d\) is obtained by DialoGPT. Then, the embeddings of the prompt sequence \(E_p\) are prepended to the \(E_d\) as a whole input embedding matrix. Overall, the PLM loss is calculated as:

\[
L_{PLM} = -\sum_{t=1}^{T} \log p_{\theta_1, \theta_2, \varphi}(y_t | y_{<t}, d, p_{att}, p_{task})
\]

where \(T\) is the length of generated sequence, \(\varphi\) is the parameter of the PLM and is fixed. The parameters of two prompts, \(\theta_1\) and \(\theta_2\), are the only updated parameters. Therefore, the training loss \(L\) is the weighted sum of the disentanglement loss and the PLM loss:

\[
L = \alpha L_D + (1 - \alpha) L_{PLM}
\]

\(^1\)We find constructing pseudo combinations with at least one different attribute value is slightly better than with all different attributes in the experiments.

Figure 4: Overview of our evaluation model, MAE.

When the training is completed, we save all parameters of the prompt module. During the inference, the data from the test set is mapped to the representations of prompts only via the embedding matrices, where the features of the attributes seen in the training set can be transferred to the unseen combinations.

5 Method of MAE

To fill the gap in metrics for multi-attribute controllable dialogue generation, we propose a unified and efficient evaluation framework without additional large-scale labeled data, as shown in Figure 4, which converts the evaluation of each attribute to a unified text-to-text generation task, just like Gu et al. (2022). T5 (Raffel et al., 2020) is used as the base model for our work. A template is designed as discrete prompts, i.e., "The emotion/act/persona controls the response [MASK]". To alleviate the potential bias of different handcrafted patterns (Ke et al., 2022), we further add a trainable continuous-task-oriented prompt to improve stability and robustness.

Specifically, the continuous prompt sequence is prepended to the response as a prefix, which makes up the input of the encoder. Another continuous prompt sequence, the attribute values, and the template are concatenated and fed to the decoder. We take the probability of generating "yes" corresponding to [MASK] token as the controllability score. In training process, only embeddings of continuous

...
prompts are updated and the parameters of T5 are fixed. Note that our model-based evaluation approach gets rid of the reliance on golden response when tested and can be uniformly applied to various granularities of attributes.

6 Experiments

6.1 Datasets
We construct two datasets based on DailyDialog (Li et al., 2017) and ConvAI2 (Dinan et al., 2020) for compositional generalization in multi-attribute controllable dialogue response generation.

DailyDialog-CG DailyDialog is an open-domain dialogue dataset with two controllable attributes: emotion and act. Here, we treat the labels of the two attributes as an attribute combination, e.g., (surprise, inform). For dialogues, each utterance with two attribute labels is regarded as the response and all preceding texts of this utterance are considered as the corresponding dialogue history. In this way, we get 14,879 examples. We count the attribute combinations labeled in all examples, 18 of which are selected as $C_{v,train}$ and the other 6 are $C_{v,test}$. Then, the examples are divided into the training set and test set according to the combination set. We also extract 10% samples from the training set as the validation set.

ConvAI2-CG ConvAI2 is a persona-based dialogue dataset in which the persona profile of each dialogue is consisting of 4 or 5 personalized sentences. We treat each sentence as an attribute value and the sentences in the same position belong to the same attribute. The persona profile is regarded as an attribute combination, e.g., (“My mom is my best friend.”, “I’ve four sisters.”, “I believe that mermaids are real.”, “I love iced tea.”). For each dialogue, we choose the first 4 utterances as the dialogue history and the 5th utterance as the response. Consistent with the processing method of DailyDialog-CG, we select 11,566 combinations as $C_{v,train}$ and the other 873 combinations as $C_{v,test}$. After that, we obtain the corresponding training set, validation set, and test set.

The statistics about the two datasets are shown in Table 1.

6.2 Baselines
We compare our methods with several competitive baselines. The common dialogue generation models are included: (1) DialoGPT-Ori (Zhang et al., 2020); (2) FUDGE (Yang and Klein, 2021); (3) PPLM (Dathathri et al., 2020); (4) Cocon (Chan et al., 2021); (5) Fine-tuning; (6) CTRL (Keskar et al., 2019). We also implement some prompt-based methods for comparison: (1) Prompt-tuning (Lester et al., 2021); (2) CatPrompt (Yang et al., 2022). More details can be seen in Appendix B.

6.3 Evaluation Metrics
In this work, we focus on evaluating the attribute controllability and text quality for different controllable generation methods.

Attribute Controllability It aims to evaluate whether the method can generate responses constrained by multiple attributes successfully.

1. For the control of coarse-grained discrete attributes in DailyDialog-CG, we use the classification accuracy, i.e., E-ACC and A-ACC, for each attribute computed by an independently trained Roberta classifier (Liu et al., 2019), respectively.

2. For the control of fine-grained continuous attributes in ConvAI2-CG, we calculate the cosine similarity between the representations of attribute sentences and the generated response, i.e., P-SIM (Du and Ji, 2021). We also evaluate the model by measuring the consistency of attribute sentences with the generated response via a Roberta-based Natural Language Inference (NLI) model, i.e., P-NLI (Madotto et al., 2019).

3. We propose a unified model-based evaluation metric, i.e., MAE, for various granularities of attributes, the details can be seen in Section 5.

Text Quality We use the BLEUs (Papineni et al., 2002a) and METEOR (Banerjee and Lavie, 2005) to measure the match scores between generated responses and ground-truth references.

6.4 Main Results
Results on DailyDialog-CG Table 2 presents the results of controllable dialogue generation about unseen attribute combinations for DailyDialog-CG. We conduct experiments based on some strong controllable dialogue generation models and novel prompt-based methods. In general, our DCG outperforms all other baselines in terms of attribute controllability and text quality. Compared to CTRL, our model improves by 1.6%, 2.7%, 4.1% in BLEU-1, BLEU-2, METEOR for text quality, and 3.3%, 3.1%, 5.5%, 5.5% in E-ACC, E-MAE, A-ACC, A-MAE for attribute controllability. We

We will release our code after blind review.

Our DCG improves text quality and controllability. The BLEUs seem low because we adopt the same calculation as ParlAI (Miller et al., 2017), which is lower than results in (Li et al., 2017) for different smooth functions.
Table 2: The performance of compositional generalization in multi-attribute controllable dialogue generation for DailyDialog-CG. "E" and "A" denote controllable attributes of "Emotion" and "Act". "AOP", "TOP", and "DL" mean attribute-oriented prompt, task-oriented prompt, and disentanglement learning. Results are averaged over three random runs. ↑ means a higher score is better. (p < 0.01 under t-test)

<table>
<thead>
<tr>
<th>Method</th>
<th>Controllability</th>
<th>Text Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>E-ACC ↑</td>
<td>E-MAE ↑</td>
</tr>
<tr>
<td>DialoGPT-Ori</td>
<td>50.36</td>
<td>60.46</td>
</tr>
<tr>
<td>FUDGE</td>
<td>60.10</td>
<td>64.29</td>
</tr>
<tr>
<td>PPLM</td>
<td>51.57</td>
<td>56.87</td>
</tr>
<tr>
<td>CoCon</td>
<td>52.79</td>
<td>59.99</td>
</tr>
<tr>
<td>Fine-tuning</td>
<td>62.74</td>
<td>66.77</td>
</tr>
<tr>
<td>CTRL</td>
<td>67.34</td>
<td>69.55</td>
</tr>
<tr>
<td>Prompt-tuning</td>
<td>57.06</td>
<td>62.78</td>
</tr>
<tr>
<td>CalPrompt</td>
<td>60.91</td>
<td>66.50</td>
</tr>
<tr>
<td>DCG (ours)</td>
<td>70.66</td>
<td>72.61</td>
</tr>
<tr>
<td>DCG w/o AOP (Prompt-tuning)</td>
<td>57.06</td>
<td>62.78</td>
</tr>
<tr>
<td>DCG w/o TOP</td>
<td>66.80</td>
<td>68.02</td>
</tr>
<tr>
<td>DCG w/o DL</td>
<td>60.41</td>
<td>64.57</td>
</tr>
</tbody>
</table>

Table 3: The performance of compositional generalization in multi-attribute controllable dialogue generation for ConvAI2-CG. "P" denotes controllable attribute of "Persona". Results are averaged over three random runs. ↑ means a higher score is better. (p < 0.01 under t-test)

<table>
<thead>
<tr>
<th>Method</th>
<th>Controllability</th>
<th>Text Quality</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>P-SIM ↑</td>
<td>P-NLI ↑</td>
</tr>
<tr>
<td>DialoGPT-Ori</td>
<td>60.16</td>
<td>72.47</td>
</tr>
<tr>
<td>PPLM</td>
<td>59.90</td>
<td>75.98</td>
</tr>
<tr>
<td>Fine-tuning</td>
<td>65.48</td>
<td>69.50</td>
</tr>
<tr>
<td>CTRL</td>
<td>65.20</td>
<td>77.65</td>
</tr>
<tr>
<td>Prompt-tuning</td>
<td>64.84</td>
<td>74.30</td>
</tr>
<tr>
<td>DCG (ours)</td>
<td>69.03</td>
<td>81.20</td>
</tr>
<tr>
<td>DCG w/o AOP (Prompt-tuning)</td>
<td>64.84</td>
<td>74.30</td>
</tr>
<tr>
<td>DCG w/o TOP</td>
<td>67.35</td>
<td>78.50</td>
</tr>
<tr>
<td>DCG w/o DL</td>
<td>68.25</td>
<td>79.00</td>
</tr>
</tbody>
</table>

also find the FUDGE and PPLM, two methods based on the decoding strategy, perform poorly here, especially in text quality, which illustrates the incompatibility of these decoding strategies for combinatorial generalization. Besides, as observed, Catprompt is a relatively well-performing prompt-based baseline, but it is still far worse than our method. This is because it directly concatenates all trained single-attribute prompts as the multi-attribute prompt for test. This inconsistency between training and testing stages decreases the performance. Different from these methods, our method optimizes the language modeling loss only based on discrete prompts for attribute combination and continuous task-oriented prompt, which can focus on the features of multiple attributes at the same time also during the training and achieve a better transfer via a learnable mapping.

Besides, we also concern whether DCG benefits from attribute-oriented prompt, task-oriented prompt, and disentanglement learning. We find that DCG w/o AOP is the same with Prompt-tuning and it performs poorly in attribute controllability, which shows attribute-oriented prompt plays an important role in guiding the model to focus on the controlled information. After removing the task-oriented prompt, the DCG w/o TOP decreases to 19.18%, 6.74%, and 15.63% on text quality, but still maintains high controllability. It proves task-oriented prompt helps improve text quality. Besides, after removing disentanglement learning, the DCG w/o DL drops significantly, which shows disentanglement learning effectively disentangles attribute combinations and improves the ability of compositional generalization.

Results on ConvAI2-CG Table 3 presents the results of generalization on unseen attribute combinations for ConvAI2-CG. Due to the diversity of attribute values and attribute combinations, it is very difficult to implement CatPrompt in ConvAI2-CG. Therefore, we remove this baseline. We also remove FUDGE and Cocon for their poor generation quality and slow decoding speed, which is shown in Table 2 and Table 5. We can observe that the trend of overall performance is consistent with that of DailyDialog-CG. Compared to CTRL, our model achieves a great improvement in attribute controllability and text quality, which proves the generality of our methods on the coarse-grained discrete attribute control and fine-grained contin-
Figure 5: Comparison of performance for Fine-tuning, CTRL, and DCG on seen and unseen multi-attribute combinations for DailyDialog-CG in terms of E-ACC, A-ACC, BLEU-1, and BLEU-2.

Table 4: Pearson ($r$), Spearman ($\rho$), and Kendall ($\tau$) correlations of attribute controllability evaluation metrics on DailyDialog-CG and ConvAI2-CG. "T1" and "T2" denote the Template 1 and Template 2.

7 Qualitative Analysis

7.1 Comparison between Seen and Unseen Attribute Values

Figure 5 displays the comparison of the performance on seen and unseen attribute combinations for DailyDialog-CG. We report the controllability metrics, E-ACC (emotion) and A-ACC (act), and the BLEUs of the Fine-tuning, CTRL, and our DCG. The top of each box denotes the result of seen attribute combinations and the bottom represents unseen attribute combinations. We find all methods achieve significantly superior performance on seen attribute combinations than on unseen combinations. For example, CTRL achieves 71.27% E-ACC and 43.15% A-ACC on seen attribute combinations but drops to 67.34%(-3.93) and 33.50%(-9.65) on unseen combinations. It strongly proves previous methods suffer from the difficulty of compositional generalization for the multi-attribute controllable dialogue generation. However, we find our proposed DCG can greatly alleviate this gap. The DCG has a smaller drop of 0.41% and 0.11% for E-ACC and A-ACC, and it also outperforms CTRL on both controllability and text equality of unseen attribute combinations. The results confirm the effectiveness of our method for transferring seen attributes to unseen combinations. We find CTRL achieves a higher A-ACC on seen combinations but a lower score on unseen combinations than Fine-tuning, which demonstrates directly adding control codes may cause overfitting to seen attribute combinations.

7.2 Correlation Results on Metrics

Following Guan and Huang (2020), we adopt Pearson ($r$), Spearman ($\rho$), and Kendall ($\tau$) correlation coefficients between our proposed automatic metrics, MAE, and human judgments (details can be seen in Appendix E) to measure the quality of different metrics. Table 4 shows the overall results on the controllability of coarse-grained discrete attributes, emotion and act, and the fine-grained continuous attributes, persona description. We can observe that our MAE outperforms classic metrics, E-ACC, A-ACC, P-SIM, and P-NLI, by a large mar-
attribute prompts makes the model only focus on the latter attribute (act), i.e., position sensitive, so that the CatPrompt cannot distinguish different combinations with the other attribute (emotion). Therefore, it’s hard for CatPrompt to learn multi-attribute compositional generalization. In Figure 7(b), We find that DCG w/o DL can distinguish different multi-attribute combinations to some extent. However, the combinations of different attribute values are tightly entangled, such as (a0, b2) and (a4, b1). Figure 7(c) shows that our DCG has a close distribution with prompts of the same attribute value, i.e., (a0, b0), (a0, b1), (a0, b2), and a sparse distribution with prompts of different attribute values, e.g., (a0, b2) and (a4, b1). It proves our DCG can disentangle attribute combinations and learn relations between different attributes. Furthermore, DCG learns generalization capability from seen attributes to unseen combinations. For example, (a2, b1) -> (a0, b1) (unseen path) is equal to (a2, b0) -> (a0, b0) (seen path). The results confirm that our proposed attribute-oriented prompt outperforms the models that learn an independent prompt for each attribute value. The shared embedding mapping helps learn attribute concepts from seen values to unseen combinations.

7.4 Few-shot Learning

To study the effect of few-shot learning, we randomly select a ratio of original training data from DailyDialog-CG and show the results in Figure 6. Note that we keep the original test set fixed for fair comparison. As the size of training data decreases, the performance of both CTRL and DCG presents a drop trend and our DCG model is consistently better than CTRL, which confirms our model has strong capability for multi-attribute controllable dialogue generation.

8 Conclusion

In this paper, we study the compositional generalization for multi-attribute controllable dialogue generation. We propose a prompt-based disentangled controllable dialogue generation model which generates attribute-specific prompt vectors from control codes and uses a disentanglement loss to disentangle different attributes. Further, we develop a unified reference-free evaluation framework, MAE, for multi-attribute generation with different levels of granularities. Experiments and analysis show our method achieves better text quality and controllability scores. Moreover, our proposed MAE has a higher correlation with human judgments for evaluation on CDG.
9 Statement of Ethical Considerations

Controllable dialogue generation (CDG) is an essential task in Natural Language Processing (NLP) and has been widely studied for decades, which aims to guide dialogue generation toward the desired attributes such as emotions, acts, and personas. In the open-domain dialogue scenario, CDG can generate emotional and diverse responses to enhance the user’s sense of participation. In the task-oriented dialogue scenario, CDG can generate responses that meet the user’s needs according to the user’s intent. However, most previous works focus on single-attribute generation where there is only one attribute label like happiness in emotion and pay less attention to the multi-attribute generation, which is a more practical setting. Different from single-attribute, the control signal of the multi-attribute generation is a combination of multiple values from different attributes, which faces the challenge of lacking sufficient annotated attribute-specific data. Therefore, we explore the compositional generalization for multi-attribute controllable dialogue generation where a model could learn from seen attribute values and generalize to unseen combinations. We also design a novel and general reference-free evaluation framework to unify the evaluation of different granularity attributes. The experimental results prove the effectiveness of our model and evaluation framework. Besides, there is no huge biased content in the datasets and the models. If the knowledge base is further used, the biased content will be brought into the generated responses, just like biased content posted by content creators on the Web which is promoted by a search engine. To prevent the technology from being abused for disinformation, we look forward to more research effort being paid to fake/biased/offensive content detection and encourage developers to carefully choose the proper dataset and content to build the knowledge base.

References


Yiwei Lyu, Paul Pu Liang, Hai Pham, Eduard Hovy, Barnabás Póczos, Ruslan Salakhutdinov, and Louis-Philippe Morency. 2021. **Styleptb**: A compositional benchmark for fine-grained controllable text style transfer.


Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott,


A  Prompt Visualization

In this section, we give the prompt visualization in Figure 7.

B  Baselines

DialoGPT-Ori: Proposed by (Zhang et al., 2020), this model is a dialogue generative pre-trained transformer. Here, we use the original DialoGPT for open-domain dialogue generation. DialoGPT is the backbone for all other baselines except CoCon.

Fine-tuning: We use dialogue history in datasets to fine-tune the DialoGPT for dialogue generation.

CTRL: Proposed by (Keskar et al., 2019), this method provides attribute control codes for a language model trained from scratch. We concatenate multi-attribute control codes with dialogue history to fine-tune the DialoGPT.

CoCon: Proposed by (Chan et al., 2021), this method uses a content input to control an GPT’s output text at a fine-grained level.

PPLM: Proposed by (Dathathri et al., 2020), this method is a gradient-based baseline that uses a plug-and-play language model (PPLM) to guide the language model. We train a joint classifier of emotion and dialogue act on DailyDialog-CG.

Noted that the attribute classifiers of PPLM can not directly generalize to unknown attribute combinations, so we use both training data and test data to train the attribute classifiers. We use the bag-of-words attribute model which encodes persona profile to control the DialoGPT on ConvAI2-CG.

FUDGE: Proposed by (Yang and Klein, 2021), this method is a weighted decoding baseline which uses a future discriminator for generation (FUDGE) to guide the DialoGPT. We train a joint discriminator that takes the dialogue history and the current response as input and predicts the attribute combination of emotion and dialogue act on DailyDialog-CG.
We fine-tune multi-attribute prompts for dialogue (Loshchilov and Hutter, 2019) and a linear learn-

experiments, we set the number of training epochs to 36 minutes per epoch on ConvAI2-CG. For all ex-

ber of Pseudo Combinations is set to 8 and 6 for

lengths are set to 6 and 100, the disentanglement

are set to 50 and 150, the attribute-oriented prompt

Face default setup, we use an AdamW optimizer

truncated to 512 tokens. Following the Hugging-

Our implementation is based on the Hugging Face

C Implementation Details

Our implementation is based on the Hugging Face

Transformer models\(^5\). DialoGPT\_Small is used

as a backbone and the input sequence length is

truncated to 512 tokens. Following the Hugging-

Face default setup, we use an AdamW optimizer

(Loshchilov and Hutter, 2019) and a linear learning

rate scheduler with an initial rate of 7.5 \cdot 10^{-5},

and the batch size is set to 8. The prompt lengths

are set to 50 and 150, the attribute-oriented prompt

lengths are set to 6 and 100, the disentanglement

loss weight is set to 0.1 and 0.03, and the num-

ber of Pseudo Combinations is set to 8 and 6 for

DailyDialog-CG and ConvAI2-CG, respectively.

Our model is trained on Tesla V100 machines, tak-
ing 24 minutes per epoch on DailyDialog-CG and

36 minutes per epoch on ConvAI2-CG. For all ex-

periments, we set the number of training epochs to

30. At the decoding phase, we use a greedy search

\(^5\)https://github.com/huggingface/transformers

and max generated tokens of 150.

D Inference Efficiency

We compare the average inference efficiency of

our methods with the baselines. As we can ob-

serve from Table 5, the inference speed of PPLM,

FUDGE, and CoCon is far slower than the original

GPT-2 model. Prompt-based methods are much

faster than that decoding strategy based methods.
The inference speed of our method is close to the

original DialoGPT methods. As shown in Table

6, with the growth of attribute combinations, the

trainable parameters of CatPrompt increase rapidly,

from 0.84M to 224M, which even exceeds the

117M trainable parameters of full DialoGPT. While

our method achieves better results with a lower

number of trainable parameters on DailyDialog-

CG and ConvAI2-CG.

Table 5: The decoding speed of different models, which

takes the decoding speed of the model relative to the

Prompt as a metric.

<table>
<thead>
<tr>
<th>Method</th>
<th>Decoding Speed $\uparrow$</th>
</tr>
</thead>
<tbody>
<tr>
<td>DialoGPT-Ori</td>
<td>1.1837x</td>
</tr>
<tr>
<td>FUDGE</td>
<td>0.0041x</td>
</tr>
<tr>
<td>PPLM</td>
<td>0.0006x</td>
</tr>
<tr>
<td>CoCon</td>
<td>0.0044x</td>
</tr>
<tr>
<td>Fine-tuning</td>
<td>1.1347x</td>
</tr>
<tr>
<td>CTRL</td>
<td>1.1673x</td>
</tr>
<tr>
<td>Prompt-tuning</td>
<td>1.0000x</td>
</tr>
<tr>
<td>CatPrompt</td>
<td>1.0408x</td>
</tr>
<tr>
<td>DCG (ours)</td>
<td>1.0490x</td>
</tr>
<tr>
<td>DCG w/o DL</td>
<td>1.0122x</td>
</tr>
</tbody>
</table>

Figure 7: Visualization of prompts from different models on DailyDialog-CG. Each dot denotes the prompt

embeddings of a multi-attribute combination \((a^*, b^*) | a^* \in A, b^* \in B\), where A is the attribute Emotion and B is

the attribute Act. The same color represents that two dots have the same value of Emotion and different shaped dots

represent seen/unseen combinations. For clarity, we leave out some outliers.
Table 6: Number of parameters used for different models. Trainable parameters is the number of parameter used for training in models. Percent Trainable is the ratio of trainable parameters to original GPT-2.

<table>
<thead>
<tr>
<th>Model</th>
<th>DailyDialog-CG</th>
<th>ConvAI2-CG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Trainable Parameters</td>
<td>Percent Trainable</td>
</tr>
<tr>
<td>Fine-tuning CTRL</td>
<td>117M</td>
<td>100%</td>
</tr>
<tr>
<td>Prompt-tuning CatPrompt</td>
<td>0.13M</td>
<td>0.11%</td>
</tr>
<tr>
<td>DCG (ours)</td>
<td>0.66M</td>
<td>0.56%</td>
</tr>
<tr>
<td>DCG w/o DL</td>
<td>0.66M</td>
<td>0.56%</td>
</tr>
</tbody>
</table>

Table 7: Human evaluation on controllability and text quality for DailyDialog-CG and ConvAI2-CG. Emo., Act., and Per. are the attributes of emotion, act, and persona. Flu. and Rel. are the fluency and context relevancy.

**E Human Evaluation**

To validate the good performance of DCG, we further deploy a set of human evaluations to compare the controllability and text quality between several methods. We randomly sample 100 examples from two datasets and collect the corresponding generated responses of CTRL, DCG, and DCG w/o DL. For the controllability, 5 human annotators are invited to evaluate on a scale of 1-3, where score 1 means that the generated response is completely inconsistent with the expected attribute label, score 2 denotes that the generated response has the same meaning as the expected attribute label, but no explicit attribute-related words, and score 3 means that the generated response contains some clear attribute words. For the text quality, we ask the annotators to evaluate the fluency and context relevance of the generated responses on a scale of 1-5, where a higher score indicates better quality. The inter-annotator agreement on the controllability and text quality between several methods is high, which is because the models fine-tuned on contextualized language backbones can generate fluent sentences with relevant information. For controllability, our DCG achieves better performance than CTRL both on the coarse-grained discrete attributes and fine-grained continuous attributes, which suggests that our shared prompt mapping can learn the attribute concepts from seen attribute values to unseen attribute combinations and is useful for diverse attributes. Besides, when removing the disentanglement learning, the scores of our DCG w/o DL drop significantly, which further shows the effectiveness of the combination disentanglement to improve the generation ability.

**F Effect of Model Parameters**

**Prompt Length** Figure 8 (a) displays the effect of overall prompt lengths of $E_p$. Since the length of attribute-oriented prompt is fixed to the number of control code, we change the length of the task-oriented prompt. We find that our DCG achieves superior performance when the prompt length is between 20 and 100, and gets the best scores when the prompt length is 50. The DCG outperforms the strong baseline CTRL by the 3.19% (averaged) for MAE and 2.16% (averaged) for BLEUs but uses only 56% trainable parameters of CTRL, which verifies the effectiveness and robustness of our method.

**Weight of Disentanglement Loss** Figure 8 (b) shows the effect of different weight ratios $\alpha$ for the disentanglement loss $L_D$. We observe that $\alpha \in (0.05, 0.15)$ achieves consistent improvements than CTRL and we take $\alpha = 0.10$ in all experiments.

**Number of Pseudo Combinations** Figure 8 (c) shows the effect of the number of pseudo combinations in the disentanglement loss. We find a larger number will improve the controllability of
Figure 8: Effect of prompt length, disentanglement loss weight, and number of pseudo combinations for DailyDialog-CG. The dotted lines denote the performance of CTRL. We report the MAE and BLEU scores for all settings.

<table>
<thead>
<tr>
<th>Metrics</th>
<th>DailyDialog-CG</th>
<th>ConvAI2-CG</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Emotion</td>
<td>Act</td>
</tr>
<tr>
<td></td>
<td>Pearson</td>
<td>Spearman</td>
</tr>
<tr>
<td>CTRLEval</td>
<td>0.6927</td>
<td>0.6994</td>
</tr>
<tr>
<td>MAE</td>
<td>0.6821</td>
<td>0.7500</td>
</tr>
</tbody>
</table>

Table 8: Pearson ($r$), Spearman ($\rho$), and Kendall ($\tau$) correlations of attribute controllability in DailyDialog-CG and ConvAI2-CG. We use the attribute relevance of CTRLEval as the controllability score.

G Case Study

Figure 9 shows two examples from Dailydialog-CG and ConvAI2-CG, respectively. For example one in the DailyDialog-CG, the CTRL generates the word "great", showing that the generated response is emotionally controllable. However, both sentences in the response are declarative sentences, which does not control the act question. As observed, the response generated by our DCG contains the word "Wow", which strongly expresses the emotion of happiness. Besides, a question sentence is also generated. Example two in ConvAI2-CG needs to control 5 attributes, of which the golden response contains 2 attributes. The CTRL only controls "like to skate", while our DCG controls "like to write poetry and skate", which is highly consistent with the golden response. Compared with previous models, our model addresses many difficult issues in compositional generalization for multi-attribute controllable dialogue generation. With an attribute-oriented prompt and a task-oriented prompt, our method learns attribute concepts from seen attribute values to unseen attribute combinations. Through a disentanglement learning, some artificial-constructed unseen pseudo combinations are injected into the training process, which greatly improves the generalization ability of our model.

H Comparison with CTRLEval

Automatic evaluation metrics are important for text generation tasks, including reference-based like BLEU (Papineni et al., 2002b), ROUGE (Lin, 2004), BERTScore (Zhang* et al., 2020) and unreferenced like perplexity (Brown et al., 1992), discriminator scores (Dathathri et al., 2020), BARTScore (Yuan et al., 2021). To evaluate controllability, (Dathathri et al., 2020; Yang and Klein, 2021) trained an attribute classifier to predict the probability using labeled external data, which is hard to multi-attribute controllable generation. As a concurrent work, CTRLEval (Ke et al., 2022) proposes an evaluation method for controllable text generation. Different from our MAE, CTRLEval uses handcrafted prompts to evaluate attribute relevance. However, handcrafted prompts are hard to construct for new tasks and cause generation bias. In contrast, our MAE uses a learnable soft prompt based on PLMs to enhance the generalization capability and robustness. We also provide a performance comparison in Table 8. Results show
Example one in DailyDialog

Dialogue History:
A: Smells good! What's for breakfast? 
B: Well, since we are getting up so late, I decided to make a big breakfast!
A: Nice! Brunch!
B: Kind of, I made scrambled and soft boiled eggs, some French toast and buttermilk pancakes!

Controllable Attribute Combination:
(happiness, question)

Golden Response
Wow! You really went all out! Did you make any coffee?

CTRL
That sounds like a great breakfast! I'll make a big omelet, toast, and buttermilk pancakes!

DCG (ours)
Wow! You really went all out with it! Did you make any coffee?

Example two in ConvAI2

Dialogue History:
A: Hi! how are you doing today?
B: I am doing good and yourself?
A: I am doing alright. What are some of your hobbies?
B: I love the outdoors and camping

Controllable Attribute Combination:
I sometimes write poetry. My favorite color is black. I sing in a punk band. I like to skate. I want to visit the uk.

Golden Response
That sounds fun. I like to write poetry and to go ice stating haha.

CTRL
Just like to skate and ski.

DCG (ours)
That is cool. I like to write poetry and skate.

Figure 9: Case study for two examples from DailyDialog-CG and ConvAI2-CG. We present the dialogue history, its corresponding controllable attribute combination, golden response, CTRL prediction, and prediction of our DCG.
Table 9: The performance of CTRL and DCG for ConvAI2-CG when the number of attributes varies. We train models with 4 attributes and inference with 5 attributes. Results are averaged over three random runs. ↑ means a higher score is better. (p < 0.01 under t-test)