Multi-Attribute Controlled Text Generation with Contrastive-Generator and External-Discriminator

Guisheng Liu[1], **Yi Li**[1*], **Yanqing Guo**[1] , **Xiangyang Luo**[2], **Bo Wang**[1]
[1]Dalian University of Technology
[2]Information Engineering University
lgs0000@mail.dlut.edu.cn, {liyi, guoyq}@dlut.edu.cn
xiangyangluo@126.com, bowang@dlut.edu.cn

Abstract

Though existing researches have achieved impressive results in controlled text generation, they focus mainly on single-attribute control. However, in applications like automatic comments, the topic and sentiment need to be controlled simultaneously. In this work, we propose a new framework for multi-attribute controlled text generation. To achieve this, we design a contrastive-generator that can effectively generate texts with more attributes. In order to increase the convergence of the text on the desired attributes, we adopt an external-discriminator to distinguish whether the generated text holds the desired attributes. Moreover, we propose top-n weighted decoding to further improve the relevance of texts to attributes. Automated evaluations and human evaluations show that our framework achieves remarkable controllability in multi-attribute generation while keeping the text fluent and diverse. It also yields promising performance on zero-shot generation.

movie positive: In the end Hitchcock brings one of those rare Hollywood thrillers that I have been craving for a long time: the suspense thriller. It is one of my favorite genres and movies because you can always be sure that something exciting will happen in the ending (and sometimes it does!). I think it is also a classic Hitchcock masterpiece.
automobile negative: In the past sixty years, GM cars have gotten worse, because GM engineers were worse. Now they are just as bad as Ford's. But GM still makes good cars. I'm sorry but car makers need to go back to making things people want, like lawnmowers.It's a big step backwards.

Table 1: Examples of the text generated by our framework. The boldfaced words indicate the desired topic and sentiment. The underlined words are the prompts. The words related to the topic are highlighted in blue, and the words related to the sentiment are highlighted in red.

1 Introduction

Large pre-trained language models (LM) have achieved impressive performance in natural language generation tasks (Radford et al., 2018; Yang et al., 2019; Radford et al., 2019; Lewis et al., 2020). However, they suffer from the limitation of lacking controllability in practical applications, for their generated texts may have toxic meanings or undesired topics. Hence how to control the text generation with desired attributes as topic, sentiment, style, etc., has been further studied.

To effectively control the desired attribute of the generated text, class-conditional language models (CC-LM) such as Conditional Transformer Language (CTRL) model (Keskar et al., 2019) generate texts conditioned on control codes. But the controlled ability are restricted since the control codes are only used at the beginning of the generation process. The Plug and Play Language Models (Dathathri et al., 2020) directly update the gradient of a classifier to generate the conditioned text without retraining or finetuning the language model. While being flexible, this kind of method is computationally expensive and leads to less fluent texts. Recently, (Yu et al., 2021) introduces an alignment function to the language model so that it can generate texts with target attributes. Future Discriminator for Generation (Yang and Klein, 2021) trains a classifier to predict the probability of the desired attribute. However, all these methods are aimed at single-attribute control, making them insufficient to deal with application scenarios that need multi-attribute control. Taking the automatic comment system as an instance, it requires to control the

[*] Corresponding author

Proceedings of the 29th International Conference on Computational Linguistics, pages 5904–5913
October 12–17, 2022.

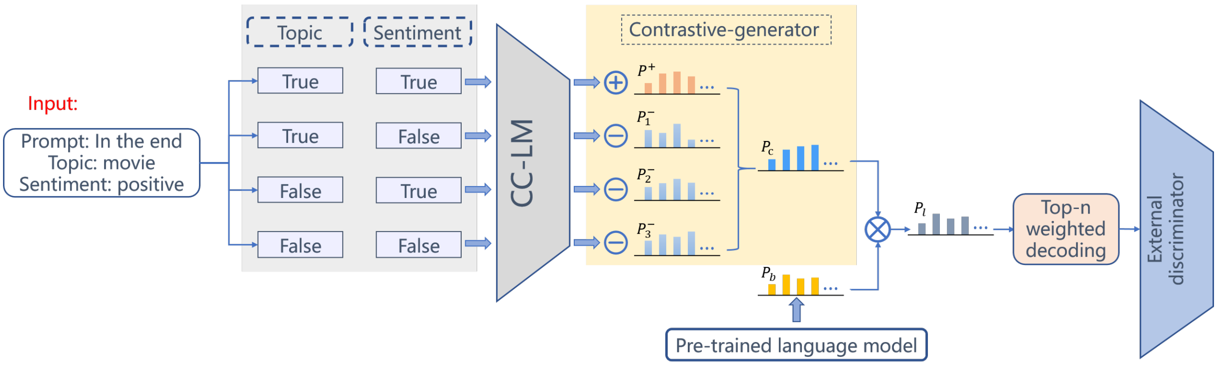

Figure 1: Illustration of our framework. Three negative samples and one positive sample are used to calculate the classification probability in CC-LM. The probability multiplies with unconditioned probability that is generated by a pre-trained language model. Through our proposed top-n weighted decoding, the external-discriminator (on the right) is used to discriminate the desired attribute of the text.

generated text with topic as well as sentiment to encourage user engagement and interactions.

In light of the problem, we propose a multi-attribute controlled framework that can effectively control topic and sentiment of the text at the same time. Although there are previous researches (Dathathri et al., 2020; Yu et al., 2021; Goswamy et al., 2020) about multi-attribute controlled text generation, they only conceptually raise the task but focus mainly on single-attribute control. The challenges of multi-attribute control lie in 1) the fusion of attributes and 2) the increased categories to be generated. To generate texts with desired topic and sentiment, we design a contrastive-generator with three negative samples in contrast to one positive sample. Since the generator may generate texts with different categories, we train an external-discriminator to increase the convergence of texts on desired attributes. In the decoding phase, a top-n weighted decoding is proposed to improve the ability of controlled text generation.

Table 1 shows the texts generated by our framework, where the texts achieve desired attributes control without losing its fluency. Zero-shot generation, which is a more challenging task, aims at generating unseen text from the seen text. Due to the top-n weighted decoding and contrastive-generator trained with external-discriminator, our framework can generate texts with other desired attributes besides the training attributes, accomplishing zero-shot generation.

We summarize the contributions of this work as follows:

- Different from existing works, we aim at multi-attribute controlled text generation, which is not only more challenging but also more practical in real-life applications as the automatic comment system.

- We propose a contrastive-generator trained with an external-discriminator to effectively generate texts with desired attributes. A top-n weighted decoding is also designed to further improve the relevance between the texts and the desired attributes.

- We conduct extensive experiments to show that our method can generate texts with desired sentiment and topic without sacrificing the linguistic quality. In addition, our framework can be generalized to new control codes and achieve promising performance on zero-shot generation.

2 Related Work

Given a control code a, the purpose of controlled text generation is to generate text x by calculating the probability of $p(x|a)$. There are mainly two categories: the first retrains language model with control codes, while the second changes the weight of the specific words for controlled text generation.

Models trained or fine-tuned (Keskar et al., 2019; Xu et al., 2020; Fang et al., 2021) on a large number of conditioned codes can achieve remarkable effectiveness for controlled text generation. However, the large training data and the computation cost are the heavy burdens. Methods with a smaller LM (Krause et al., 2021; Yang and Klein, 2021; Liu et al., 2021) to guide generation from large LM can generate text for sentiment control or topic control. GeDi (Krause et al., 2021) uses Bayes rule to compute classification likelihoods of tokens, and its generative discriminator performs well on controlled text generation. Similarly, a future discriminator (Yang and Klein, 2021) is used to determine whether the desired attribute will appear in the future text. Expert LM and anti-expert LM (Liu et al., 2021) are utilized to reweight the predictions of the large LM. Considering the promising results of the discriminator, a semantic discriminator (Betti et al., 2020) is used to discriminate the coherence with external conditioning. To get a better attribute representation, (Yu et al., 2021; Xie et al., 2022) introduces a new alignment function for sentiment control. A sentence-level emotion classifier (Zhang and Wang, 2021) is adopted to generate comments with the target emotion. The methods of retraining language models tend to produce fluent texts, but they need large labeled datasets for training.

Weighted decoding is also a useful method to control text generation with language models. These methods (Hu et al., 2019; Pascual et al., 2020, 2021), which increase the probability of tokens that are similar to the target keyword or topic, control the desired attributes of texts flexibly. Metropolis-Hastings sampling is used to generate texts with more keywords (Miao et al., 2019). But since these methods do not update LM, it will decrease the fluency of the generated texts. The following works (Dathathri et al., 2020; Sha, 2020; Duan et al., 2020; Lin and Riedl, 2021; Madotto et al., 2020) conduct constrained generation under gradient guidance. To control the sentiment better, the method in (Goswamy et al., 2020) focuses on controlling more emotion categories and emotion intensity. However, gradient-based methods may lead to more computation to calculate the word's probability. Although modifying the distribution of language models is a flexible way for controlled text generation, it will sacrifice the texts quality.

Our method draws on the above two thoughts. We design a contrastive-generator and an external-discriminator by retraining the language model to control text generation while keeping the text fluency. And we also propose top-n weighted decoding to increase the correlation of attributes.

3 Methodology

As is illustrated in Figure 1, our method is composed of three main modules. We use CTRL as CC-LM to generate one positive sample and three negative samples for the contrastive-generator. Then the contrastive-generator (subsection 3.1) outputs classification probabilities that guide the generation of the pre-trained LM. Since there are multiple attributes to be considered in multi-attribute controlled text generation, we use an external-discriminator (subsection 3.2) to estimate whether the generated text achieves the target attributes. To ensure the text fluency, the top-n weighted decoding (subsection 3.3) recalculates the probabilities of the n most probable words. The details of these modules are described in the following subsections.

3.1 Contrastive-Generator

The fusion of different attributes is one of the challenges for multi-attribute controlled text generation. To deal with the issue, we propose a contrastive-generator to generate texts with the desired topic and sentiment.

Given the desired attribute a_t and a_s, our task is to learn the probability distribution $P\left(x_{1:N} \mid a_t, a_s\right)$ where $x_{1:N}$ denotes a complete text $\left(x_1, \ldots, x_N\right)$. In particular, we use a_t to present the desired topic control code while a_s for the desired sentiment control code. The CC-LM generates a completed text $x_{1:N}$ by the following equation:

$$P\left(x_{1:N} \mid a_t, a_s\right)=\prod_{i=1}^{N} P\left(x_i \mid x_{<i}, a_t, a_s\right). \quad (1)$$

Then we refer $\mathcal{L}_g$ as the conditioned language model loss:

$$\mathcal{L}_g=-\sum_{i=1}^{N} \log P\left(x_i \mid x_{<i}, a_t, a_s\right). \quad (2)$$

The contrastive-generator aims to learn the effective representation by pulling the positive samples close and pushing apart negative samples. We use $\bar{a}_t$ as the undesired topic control code and use $\bar{a}_s$ as the undesired sentiment control code. A class-conditioned language model is adopted to get a positive sample $P\left(x_{1:N} \mid a_t, a_s\right)$ and three negative samples $P\left(x_{1:N} \mid \bar{a}\right)$ where $\bar{a} \in \left\{\left(\bar{a}_t, a_s\right), \left(a_t, \bar{a}_s\right) \left(\bar{a}_t, \bar{a}_s\right)\right\}$. Obeying Bayes rule, we compute $P\left(a_t, a_s \mid x_{1:N}\right)$ as the classification probability that guides the generation of the pre-trained LM:

$$\begin{aligned} P\left(a_t, a_s \mid x_{1:N}\right) &= \frac{P\left(a_t, a_s\right) P\left(x_{1:N} \mid a_t, a_s\right)}{\sum\limits_{a} P\left(a\right) P\left(x_{1:N} \mid a\right)} \\ &= \frac{P\left(a_t, a_s\right) \prod_{i=1}^{N} P\left(x_i \mid x_{<i}, a_t, a_s\right)}{\sum\limits_{a} P\left(a\right) \prod_{i=1}^{N} P\left(x_i \mid x_{<i}, a\right)} \end{aligned} \tag{3}$$

where $a \in \left\{\left(a_t, a_s\right), \left(\bar{a}_t, a_s\right), \left(a_t, \bar{a}_s\right) \left(\bar{a}_t, \bar{a}_s\right)\right\}$. The loss function of the contrastive-generator is

$$\mathcal{L}_c = -\log P\left(a_t, a_s \mid x_{1:N}\right). \tag{4}$$

Then we use $P\left(a_t, a_s \mid x_{1:N}\right)$ to guide the generation of the large pre-trained LM. For the generation on attribute a_t and a_s, we have:

$$\begin{aligned} P\left(x_N \mid x_{<N}, a_t, a_s\right) &= \frac{P\left(x_N, a_t, a_s \mid x_{<N}\right)}{P\left(a_t, a_s \mid x_{<N}\right)} \\ &= \frac{P\left(x_N \mid x_{<N}\right) P\left(a_t, a_s \mid x_{1:N}\right)}{P\left(a_t, a_s \mid x_{<N}\right)}. \end{aligned} \tag{5}$$

Since a_t and a_s are given and the sentence $x_{1:N-1}$ has been calculated, we can draw the conclusion that $P\left(a_t, a_s \mid x_{<N}\right)$ is a constant. So we simplify the Equation 5 by the following:

$$\begin{aligned} &P\left(x_N \mid x_{<N}, a_t, a_s\right) \propto \\ &\qquad P\left(x_N \mid x_{<N}\right) P\left(a_t, a_s \mid x_{1:N}\right)^{\alpha} \end{aligned} \tag{6}$$

where α is a a hyper-parameter that controls the weight of the desired attribute. On the right, the first part is essentially a language model. The second part can be calculated by Equation 3, which is essentially the desired attribute probability of the text calculated by the contrastive-generator.

3.2 External-Discriminator

Since our work aims at multi-attribute controlled text generation, it requires to take more than one category into consideration. We propose an external-discriminator to distinguish whether the text holds the desired attributes, which further increases the convergence of the text on the desired attributes.

The external-discriminator transforms its input into an embedding matrix and outputs a probability. To alleviate the computation burden, we use multi-layer bi-directional GRU as external-discriminator. Here we implement D_ϕ as the classifier to distinguish between the texts with the desired attributes and with the undesired attributes. The external-discriminator loss can be defined as:

$$\begin{aligned} \mathcal{L}_{external} = &-\left\{\left(a_t, a_s\right) \log D_\phi\left(a_t, a_s \mid x_{1:N}\right)\right. \\ &\left.+\left(1-\left(a_t, a_s\right)\right) \log\left(1-D_\phi\left(a_t, a_s \mid x_{1:N}\right)\right)\right\} \end{aligned} \tag{7}$$

where $D_\phi\left(a_t, a_s \mid x_{1:N}\right)$ is the probability predicted by D_ϕ indicating that the text $x_{1:N}$ belongs to the desired topic a_t and the desired sentiment a_s. In order to achieve a better performance on the desired attribute control, the external-discriminator tries to guide the sentence towards the desired attributes with decreasing the external-discriminator loss.

In the end, the overall loss function for our framework is a weighted sum of three loss terms:

$$\mathcal{L}_{total} = \frac{\lambda_g}{\tau}\mathcal{L}_g + \frac{\lambda_c}{\tau}\mathcal{L}_c + \frac{\lambda_e}{\tau}\mathcal{L}_{external} \tag{8}$$

where λ_* are the hyper-parameters that reflect the strength of each loss and τ is calculated by the following equation:

$$\tau = \lambda_g + \lambda_c + \lambda_e. \tag{9}$$

3.3 Top-n Weighted Decoding

Recent researches have made impressive progress in weighted decoding (Fan et al., 2018; Holtzman et al., 2019; Pascual et al., 2020, 2021). In the decoding time, we propose a top-n weighted decoding to increase the topic relevance while generating fluent texts. Through LM, we get the probabilities of all lexical words. Different from previous methods, we modify the probabilities of the n most probable choices, instead of changing each of the words.

Utilizing the vectors of words, we calculate the cosine similarity between the topic words and the n most probable candidate words. And we use the max function to increase the weight of the related words while keeping the weight of unrelated words as it is. The reason is to increase the fluency of the texts as much as possible. Let $\nu\left(\omega_{topic}\right) \in \mathbb{R}^d$ denote the topic vector, and $\nu\left(\omega'_{top-n}\right) \in \mathbb{R}^{n\times d}$ be the n vectors of n most probable words, where d is the dimension of the vector. The modified probability l_{top-n} is calculated as:

$$\begin{aligned} &l_{top-n} = l'_{top-n} + \\ &\quad \gamma \cdot \max\left(0, \cos\left(\nu\left(\omega_{topic}\right), \nu\left(\omega'_{top-n}\right)\right)\right) \end{aligned} \tag{10}$$

Method	Diversity			Fluency	Sentiment	Topic	
	Dist-1 ↑	Dist-2 ↑	Dist-3 ↑	Perplexity ↓	Prob. ↑	Prob. ↑	Acc. ↑
GPT-2 Medium	0.20	0.64	0.86	**30.75**	0.47	0.09	0.29
GeDi-sentiment	0.23	**0.71**	**0.91**	64.10	**0.90**	\	\
DAPT-sentiment	0.14	0.50	0.74	33.74	0.75	\	\
PPLM-sentiment	0.17	0.58	0.86	44.02	0.70	\	\
DEXPERTS	0.15	0.45	0.62	36.66	0.89	\	\
GeDi-topic	0.19	0.58	0.80	59.48	\	0.46	0.85
DAPT-topic	0.14	0.50	0.72	54.68	\	0.55	0.90
PPLM-topic	0.18	0.59	0.86	39.02	\	0.33	0.76
Plug-and-Blend	**0.29**	0.67	0.76	74.99	\	0.39	0.80
PPLM	0.17	0.57	0.81	80.67	0.66	0.47	0.87
CATG	0.18	0.54	0.72	51.74	0.66	0.26	0.51
Ours	0.17	0.58	0.83	32.58	**0.90**	**0.60**	**0.92**

Table 2: The result of multi-attribute controlled text generation. We use boldface to indicate the best performance. For methods of GeDi, DAPT and PPLM-sentiment(topic), we train and evaluate its topic model and sentiment model respectively.

where l'_{top-n} refers to the original probabilities and γ is a hyper-parameter that controls the weight of the modification. As $\gamma \to 0$, the effect of the weighted decoding decreases. In our experiments, we find that the value of γ works well in the range 2 - 5.

Furthermore, since the top-n weighted decoding merely adjusts n probabilities, it not only keeps the generated texts fluent but also decreases the computation cost while controlling the desired attributes.

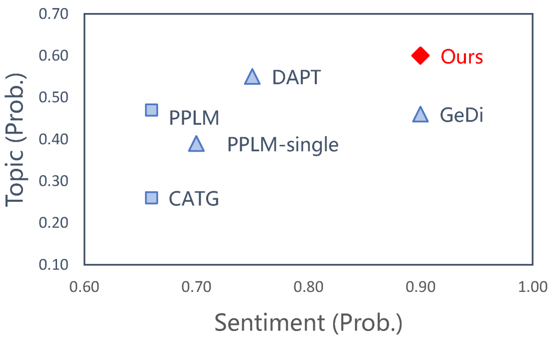

Figure 2: The analysis of sentiment and topic control. We take the sentiment mean probability as the abscissa and topic mean probability as the ordinate. The top right corner lies the best result. GeDi, DAPT and PPLM-single (showed with triangle) generate topic and sentiment texts separately. PPLM and CATG (showed with square) generate texts with topic and sentiment attributes. Our method (showed with red rhombus) surpasses both single-attribute control and multi-attribute control models.

4 Experiment

We conduct experiments on the task of multi-attribute controlled text generation (subsection 4.1) and zero-shot generation (subsection 4.2) to evaluate the performance of our framework. The ablation experiments (subsection 4.3) are also presented to analyze the importance of each module.

4.1 Multi-Attribute Controlled Text Generation

4.1.1 Evaluation

To avoid the influence of the pre-trained language model, we use GPT-2 Medium (Radford et al., 2019) as the basic language model both in our method and in the baselines. In order to evaluate the topic and the sentiment control ability of our method, we collect 500 neutral prompts that are irrelevant to the trained topics.

We use IMDb (Maas et al., 2011), OpeNER (Agerri et al., 2013) and SenTube(Uryupina et al., 2014) as the datasets for multi-attribute controlled generation. The IMDb dataset is about movie reviews with positive and negative sentiments. The OpeNER dataset is about hotel reviews that have the same two sentiments as IMDb. The SenTube dataset contains reviews in tablet and automobile domains, and for each domain we take positive and negative texts. In summary, there are four topics (movie, hotel, tablet, automobile) and two sentiments (positive and negative) during training.

Method	Fluency	Sentiment	Topic
GeDi-sentiment	3.46	4.06	\
DAPT-sentiment	4.05	3.26	\
PPLM-sentiment	3.56	3.34	\
DEXPERTS	3.88	3.94	\
GeDi-topic	3.53	\	4.02
DAPT-topic	3.67	\	3.96
PPLM-topic	3.56	\	3.20
Plug-and-Blend	3.40	\	3.66
PPLM	2.98	2.84	3.84
CATG	3.34	3.12	2.66
Ours	**4.10**	**4.14**	**4.26**

Table 3: Human evaluation of fluency and texts relevancy on the desired sentiment and topic.

animal positive:	In the past sixty years, animal welfare has increased in many countries around the world. It is an ongoing process, and we are all part of it. We can all be a part of it! And that's what I'm doing with my blog! I want to share with you my thoughts on animals, and help you make decisions about how to treat your own household animals as well as other animals.
school negative:	In a shocking finding that raises serious questions about school safety and security, researchers found that at least seven schools have experienced incidents involving armed guards or police. A report from SafeSchools.org says that between 2007 and 2014 there have been five incidents involving armed guards or police at more than 20 schools across the United States, with three resulting in fatalities. In all but two cases, it says, there was no immediate threat to students or staff.

Table 4: Zero-shot generation by our framework. Boldfaced words indicate the desired topic and sentiment. We use underlined words to show the prompts. Words related to the topic are highlighted in blue, and words related to the sentiment are highlighted in red.

In this paper, we adopt automatic evaluation as well as human evaluation to appraise the generated texts. For the automatic evaluation, we take the following four metrics into account.

- *diversity*. Diversity (Li et al., 2016) is a metric that evaluates the the diversity of the generated sentences. We report Dist-1, Dist-2 and Dist-3 by measuring the diversity of unigrams, bigrams and trigrams in the generation. A higher value indicates better diversity.

- *perplexity*. Perplexity is an automated measure of sentence fluency, lower being better. We utilize GPT-2 XL (Radford et al., 2019) to compute the perplexity of the generated text, because we use GPT-2 Medium as the pre-trained language model.

- *sentiment*. We evaluate the generations by HuggingFace's sentiment analysis classifier. The classifier achieves the accuracy of over 98% on the test data. And we obtain the mean probability from the classifier.

- *topic*. We train a topic classifier to determine whether the generated text has the desired topic attribute. The accuracy of the topic classifier is above 98%. We also report the topic accuracy and the mean probability that the text has the desired topic attribute.

4.1.2 Baseline

We compare our framework with the competitive baselines:

GPT-2 Medium: (Radford et al., 2019) To explore the influence of the pre-trained language model, we generate sentences by GPT-2 Medium as an original baseline.

PPLM: (Dathathri et al., 2020) PPLM uses gradient update to guide GPT-2 model. We retrain its discriminator to control the sentiment and topic of the text. And we evaluate its performance on single-attribute control and multi-attribute control respectively.

GeDi: (Krause et al., 2021) GeDi uses small LM as the generative discriminator to guide the generation of large LM. We separately train its topic model and sentiment model on our dataset with only topic labels or sentiment labels.

DEXPERTS: (Liu et al., 2021) DEXPERTS reweights the predictions of LM by the expert and anti-expert model. We use DEXPERTS to control the sentiment of texts for comparison.

DAPT: (Gururangan et al., 2020) DAPT shows the importance of pretraining the model towards a specific task. We use the method to generate sentiment text and topic text via training on our dataset.

CATG: (Goswamy et al., 2020) CATG controls the sentiment of sentences with a knob to influence

Desired topic	Method	Diversity			Fluency	Sentiment	Topic	
		Dist-1 ↑	Dist-2 ↑	Dist-3 ↑	Ppl. ↓	Prob. ↑	Prob. ↑	Acc. ↑
School	GeDi-topic	0.17	0.63	0.89	67.68	\	0.20	0.45
	Plug-and-Blend	0.27	0.63	0.76	67.57	\	0.18	0.50
	Ours (positive)	0.18	0.63	0.86	35.45	0.85	**0.49**	0.84
	Ours (negative)	0.16	0.60	0.86	25.23	**0.87**	0.47	0.83
Animal	GeDi-topic	0.19	**0.65**	**0.90**	67.15	\	0.29	0.73
	Plug-and-Blend	**0.28**	0.64	0.76	57.59	\	0.24	0.68
	Ours (positive)	0.17	0.62	0.87	30.09	0.78	0.38	0.88
	Ours (negative)	0.16	0.62	0.87	**23.07**	0.84	0.39	**0.92**

Table 5: The result of zero-shot generation. We use boldface to indicate the best performance.

the sentiment intensity. All the hyper-parameters are set following its original paper.

Plug-and-Blend: (Lin and Riedl, 2021) Plug-and-Blend allows multiple topic codes to generate texts. We use the model as a topic control baseline.

For all baseline methods, we generate 500 sentences for each category of topics and sentiments by our collected prompts. All the experiments are run on NVIDIA Tesla V100 GPUs. And in our framework, we configure the top-n weighted decoding with $\gamma = 4$ and $n = 50$. The values of the hyper-parameters λ_*: $\lambda_g = 0.8$, $\lambda_c = 0.2$, $\lambda_e = 1.0$.

4.1.3 Automatic Evaluation

The results are shown in Table 2. Our framework outperforms all baselines on sentiment metrics and topic metrics. It demonstrates the effectiveness of our framework on simultaneous control of both sentiment and topic. Concretely, our contrastive-generator can generate texts with multi-attribute excellently. The external-discriminator, which distinguishes the text with the desired sentiment and topic, increases the convergence of the text on the desired attributes. As is shown in Figure 2, though DAPT and GeDi train its topic and sentiment models separately, our method produces comparable or even better results in merely one model. Our framework obtains the highest mean probability and mean accuracy on the topic metrics, indicating that the top-n weighted decoding fertilizes the relevance between the texts and the desired topic effectively.

Meanwhile, texts generated by our framework acquire qualified fluency. This is because the contrastive-generator guides the basic LM without losing fluency. On the other hand, the top-n weighted decoding only modifies n words with high probability, which guarantees the maximum consistency with LM. Sentiment results on GPT-2 Medium show that our collected prompts are nearly neutral prompts that have little effects on the sentiment control. Similarly, the low topic metrics of it verify that our collected prompts are unrelated with topic. The reason why our framework is not outstanding on the diversity metrics is that our generation is under the control of sentiment and topic. And the more control leads to more limitations for generation that would hinder the diversity inherently.

Table 1 shows the texts generated by our framework. We can observe that the generated texts focus on the desired topic closely while keeping the desired sentiment. Since our training dataset are comments, our generated texts are more likely to comment on something.

4.1.4 Human Evaluation

We also conduct a human evaluation to compare the performance of baselines and our framework comprehensively. We randomly selected 20 samples from the generated texts for each method. All samples are randomly shuffled and the generation methods are completely hidden. We ask 50 annotators to evaluate the texts by the following criteria: *fluency*, *sentiment* and *topic*. Every criterion is evaluated on a scale of 1-5, where a higher score indicates better quality.

Table 3 presents the average scores of human evaluation, from which we can draw similar conclusions with the automatic evaluation. Our framework outperforms the baselines in topic and sentiment controlling while holding better fluency. We observe that GeDi has good performance on attribute control, but it can not control sentiment and topic in one sentence. Comparing with PPLM which directly updates the gradients of the pre-

Method	Diversity			Fluency	Sentiment	Topic	
	Dist-1 ↑	Dist-2 ↑	Dist-3 ↑	Perplexity ↓	Prob. ↑	Prob. ↑	Acc. ↑
Full framework	0.17	0.58	0.83	32.58	0.90	0.60	0.92
Without W	0.18	0.62	0.86	35.29	0.86	0.47	0.80
Without D	0.16	0.59	0.85	30.19	0.76	0.55	0.90
Without W,D	0.17	0.61	0.86	28.46	0.77	0.44	0.79

Table 6: Automatic evaluations of ablation study. "Without W" means that we not use top-n weighted decoding. "Without D" means that we not use external-discriminator.

trained LM, our framework has better performance on the fluency. This is because the top-n weighted decoding only changes the probabilities of the most likely n words, avoiding decreasing the text fluency significantly.

4.2 Zero-Shot Generation

We train four topics (movie, hotel, tablet, automobile) with two sentiments (positive, negative). Topics such as "school" or "animal" not appearing in the training dataset, our framework is able to generate texts with these unseen attributes. We show two examples in Table 4. Although we do not train on the two topics, our framework can effectively generate texts with the desired topic and sentiment.

We evaluate the zero-shot generation with the same metrics as the multi-attribute control. In addition, we train a topic classifier on DBPedia dataset (Zhang et al., 2015) to determine whether the generation has the desired topic attribute. The classifier achieves the accuracy of 99% on the test data.

We run experiments with zero-shot generation on the topic of "school" and "animal". For each topic, our framework generates 500 sentences with the collected 500 prompts. And we compare competitive models with Plug-and-Blend (Lin and Riedl, 2021) and GeDi (Krause et al., 2021).

The results are listed in Table 5. Our method gains better topic controlling metrics than the others while keeping the desired sentiment. It implies that the contrastive-generator generates texts effectively with unseen attributes due to its training with the external-discriminator. In addition, the top-n weighted decoding improves the relevance of the texts to the desired topic without losing the text fluency. We observe that our framework shows mediocre performance on diversity, because our framework generates texts under the control of sentiment and topic at the same time, which brings barrier to generating diverse texts. Considering that the topics are not trained, our proposed framework generalizes the pre-trained LM to generate texts with unseen categories.

4.3 Ablation

To understand the importance of each module in our framework, we perform an ablation study by training the following ablated versions: without external-discriminator, without top-n weighted decoding, without external-discriminator and top-n weighted decoding.

Table 6 presents the automatic evaluation of the ablation study. Results show that all three ablation operations will result in the decrease in attribute control performance. But since our contrastive-generator can effectively guide conditional generation by the large LM, the results about topic and sentiment still yields high values. From the result of without external-discriminator version, we observe that the topic metrics obtain relatively good results. The reason is that the top-n weighted decoding significantly improves the topic coherence. Compared to the removal of top-n weighted decoding, the full framework shows higher results of the topic and sentiment. Because the signals of the discriminator in training not only evaluate the desired attribute, but also enhance the relation between attributes and texts. In detail, from the result of without top-n weighted decoding and external-discriminator version, we notice that the average probabilities of sentiment and topic are reduced by 0.13 and 0.16 respectively. This indicates that both external-discriminator and top-n weighted decoding can effectively improve the control of sentiment and topic.

5 Conclusion and Future Work

In this paper, we propose an effective framework for multi-attribute controlled text generation. Experiments and further analysis demonstrate that the contrastive-generator and the external-discriminator perform essentially on multi-attribute generation and zero-shot generation. And the controllability of the desired attributes is further improved by our proposed top-n weighted decoding without losing the quality of texts. We also conduct the ablation experiment, showing the importance of each module. In addition to the topic and sentiment control, our framework is capable of applying to other multi-attribute control. In the future, we will generalize our model to generate texts with other attributes, e.g. writing styles and toxicity, making the generation more safer and more qualified.

6 Ethical Consideration

Since the proposed framework can be used to generate texts with more desired attributes, its generation is more like human-generated. It would benefit language generation applications on downstream tasks, such as automatic comments and chatting robots. Although automatic comments can encourage user interactions, it may mislead public opinions when it is used for malicious purposes. Moreover, we observe that the work may generate toxic texts when a negative attribute is given. Hence in the future, we will investigate how to detect toxic texts and replace the offensive words without changing the meaning of the text.

7 Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (No. 62106037, No. 62076052), in part by the Major Program of the National Social Science Foundation of China (No.19ZDA127), in part by the Fundamental Research Funds for the Central Universities (No. DUT22YG205, No. DUT20RC(3)088, No. DUT20TD110), and in part by the Open Project Program of the National Laboratory of Pattern Recognition (NLPR) (No. 202100032).

References

Rodrigo Agerri, Montse Cuadros, Sean Gaines, and German Rigau. 2013. Opener: Open polarity enhanced named entity recognition. *Procesamiento del Lenguaje Natural*, pages 215–218.

Federico Betti, Giorgia Ramponi, and Massimo Piccardi. 2020. Controlled text generation with adversarial learning. In *Proceedings of the 13th International Conference on Natural Language Generation*, pages 29–34.

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

Yuguang Duan, Canwen Xu, Jiaxin Pei, Jialong Han, and Chenliang Li. 2020. Pre-train and plug-in: Flexible conditional text generation with variational auto-encoders. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 253–262.

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

Le Fang, Tao Zeng, Chaochun Liu, Liefeng Bo, Wen Dong, and Changyou Chen. 2021. Transformer-based conditional variational autoencoder for controllable story generation. *arXiv preprint arXiv:2101.00828*.

Tushar Goswamy, Ishika Singh, Ahsan Barkati, and Ashutosh Modi. 2020. Adapting a language model for controlled affective text generation. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2787–2801.

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

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

J Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019. Improved lexically constrained decoding for translation and monolingual rewriting. 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 839–850.

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*.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. 2021. Gedi: Generative discriminator guided sequence generation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4929–4952.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of NAACL-HLT*, pages 110–119.

Zhiyu Lin and Mark O Riedl. 2021. Plug-and-blend: a framework for plug-and-play controllable story generation with sketches. In *Proceedings of the AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, volume 17, pages 58–65.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. 2021. Dexperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Andrea Madotto, Etsuko Ishii, Zhaojiang Lin, Sumanth Dathathri, and Pascale Fung. 2020. Plug-and-play conversational models. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 2422–2433.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. Cgmh: Constrained sentence generation by metropolis-hastings sampling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6834–6842.

Damian Pascual, Beni Egressy, Florian Bolli, and Roger Wattenhofer. 2020. Directed beam search: Plug-and-play lexically constrained language generation. *arXiv preprint arXiv:2012.15416*.

Damian Pascual, Béni Egressy, Clara Isabel Meister, Ryan Cotterell, and Roger Wattenhofer. 2021. A plug-and-play method for controlled text generation. *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3973–3997.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

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

Lei Sha. 2020. Gradient-guided unsupervised lexically constrained text generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8692–8703.

Olga Uryupina, Barbara Plank, Aliaksei Severyn, Agata Rotondi, and Alessandro Moschitti. 2014. Sentube: A corpus for sentiment analysis on youtube social media. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 4244–4249.

Xin Xie, Yi Li, Huaibo Huang, Haiyan Fu, Wanwan Wang, and Yanqing Guo. 2022. Artistic style discovery with independent components. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 19870–19879.

Peng Xu, Mostofa Patwary, Mohammad Shoeybi, Raul Puri, Pascale Fung, Animashree Anandkumar, and Bryan Catanzaro. 2020. Megatron-cntrl: Controllable story generation with external knowledge using large-scale language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2831–2845.

Kevin Yang and Dan Klein. 2021. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535.

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

Dian Yu, Zhou Yu, and Kenji Sagae. 2021. Attribute alignment: Controlling text generation from pre-trained language models. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2251–2268.

Linhao Zhang and Houfeng Wang. 2021. Towards controlled and diverse generation of article comments. *arXiv preprint arXiv:2107.11781*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.