CTRLEval: An Unsupervised Reference-Free Metric for Evaluating Controlled Text Generation

Pei Ke[1], Hao Zhou[2], Yankai Lin[2], Peng Li[3]*, Jie Zhou[2], Xiaoyan Zhu[1], Minlie Huang[1]†

[1]The CoAI group, DCST, Institute for Artificial Intelligence, State Key Lab of Intelligent Technology and Systems,

Beijing National Research Center for Information Science and Technology, Tsinghua University, Beijing 100084, China

[2]Pattern Recognition Center, WeChat AI, Tencent Inc., China

[3]Institute for AI Industry Research (AIR), Tsinghua University, China

`kepei1106@outlook.com`, `{tuxzhou,yankailin,withtomzhou}@tencent.com`

`lipeng@air.tsinghua.edu.cn`, `{zxy-dcs,aihuang}@tsinghua.edu.cn`

Abstract

Existing reference-free metrics have obvious limitations for evaluating controlled text generation models. Unsupervised metrics can only provide a task-agnostic evaluation result which correlates weakly with human judgments, whereas supervised ones may over-fit task-specific data with poor generalization ability to other datasets. In this paper, we propose an unsupervised reference-free metric called *CTRLEval*, which evaluates controlled text generation from different aspects by formulating each aspect into multiple text infilling tasks. On top of these tasks, the metric assembles the generation probabilities from a pre-trained language model without any model training. Experimental results show that our metric has higher correlations with human judgments than other baselines, while obtaining better generalization of evaluating generated texts from different models and with different qualities[1].

1 Introduction

Controlled text generation aims to generate texts under some control variables, including pre-specified content prefixes and attribute labels (such as sentiments and topics). Controlled text generation has been significantly advanced by large-scale pre-trained models with respect to generation quality and various control variables (Keskar et al., 2019; Dathathri et al., 2020; Yang and Klein, 2021; Liu et al., 2021a; Chan et al., 2021).

Despite the great success of these generation models, it becomes critical to evaluate the quality of generated texts accurately. Most of the existing studies adopt unsupervised and supervised metrics to measure the quality of generated texts

under different combinations of control variables (Dathathri et al., 2020; Chan et al., 2021). The evaluation is commonly conducted in a reference-free setting because it is challenging to collect sufficient high-quality references for each input of control variables in this open-ended text generation task (Dathathri et al., 2020).

However, both unsupervised and supervised metrics have shown limitations in the evaluation of controlled text generation: 1) Unsupervised metrics such as perplexity (Brown et al., 1992) can only provide task-agnostic evaluation regarding the overall quality of generated texts. However, controlled text generation tasks typically involve multiple evaluation aspects (Deng et al., 2021), including the quality of generated texts themselves and the relationship between generated texts and control variables. It is thus not surprising that existing unsupervised metrics without multi-aspect interpretability have low correlations with human judgments (Hashimoto et al., 2019). 2) Supervised metrics are commonly trained on the datasets of specific tasks to measure the corresponding aspects of generated texts (e.g., evaluating whether a generated text is accordant with the sentiment label) (Dathathri et al., 2020; Chan et al., 2021). This may cause over-fitting to task-specific data and degrade the generalization ability of metrics (Garbacea et al., 2019), thereby giving unstable evaluation of generated texts from different models or with different qualities (Guan and Huang, 2020).

To deal with the above issues, we propose an unsupervised reference-free metric called *CTRLEval* for evaluating controlled text generation models. This metric performs evaluation from different aspects without any training on task-specific data. Specifically, we formulate the evaluation of each aspect into "fill-in-the-blank" tasks whose input and output patterns can be designed based on the definition of the aspect. Then, we utilize a pre-trained model whose pre-training task is text in-

---

[1]The data and codes are available at `https://github.com/thu-coai/CTRLEval`.

filling (such as PEGASUS (Zhang et al., 2020a)) as our base model, and fuse the generation probabilities from these "fill-in-the-blank" tasks as the evaluation result. To alleviate the potential bias caused by the task design (Zhao et al., 2021), we devise multiple text infilling tasks for each aspect and use the weighted sum of all the results as the final score. In this paper, we consider three aspects which are commonly used to measure the performance of controlled text generation models, including coherence (Yuan et al., 2021), consistency (Rashkin et al., 2020), and attribute relevance (Dathathri et al., 2020). These evaluation aspects cover both the quality of generated texts and the relationship between generated texts and different control variables, which can provide a comprehensive evaluation result for controlled text generation. Experimental results show that our metric can maintain the generalization ability and achieve stable performance faced with model drift and quality drift.

Our main contributions are as follows:

- We propose an unsupervised reference-free metric called CTRLEval for evaluating controlled text generation. This metric formulates three evaluation aspects (i.e., coherence, consistency, and attribute relevance) into multiple text infilling tasks, and utilizes the ensemble of generation probabilities from a pre-trained language model as the evaluation results.

- We conduct experiments on two benchmark tasks including sentiment-controlled and topic-controlled text generation based on our collected evaluation set. Experimental results show that our proposed metric has higher correlations with human judgments, while obtaining better generalization of evaluating generated texts from different models and with different qualities.

2 Related Work

2.1 Controlled Text Generation

Early studies on controlled text generation adopt attribute label embeddings (Ficler and Goldberg, 2017; Zhou et al., 2018) or latent variables (Hu et al., 2017; Ke et al., 2018; Zhou and Wang, 2018) to learn the complex relationship between control variables and generated texts. With the development of large-scale generative pre-trained

models, it is costly to re-train or fine-tune pre-trained models on the corpora with attribute annotations (Keskar et al., 2019). Recent works resort to decoding-time methods and directly make pre-trained models generate texts towards desired attributes during inference, including PPLM (Dathathri et al., 2020), GeDi (Krause et al., 2020), FUDGE (Yang and Klein, 2021) and DEXPERTS (Liu et al., 2021a). These works rely heavily on human evaluation because existing reference-free metrics including unsupervised and supervised ones are shown to have evident limitations for evaluating controlled text generation (Dathathri et al., 2020).

2.2 Evaluation Metric for Text Generation

Automatic evaluation metrics are important for natural language generation tasks, which can be simply divided into referenced, reference-free (also known as unreferenced) and hybrid metrics: 1) Referenced metrics usually measure the relevance between generated texts and reference texts via lexicon overlap (such as BLEU (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005) and ROUGE (Lin, 2004)) or embedding similarity (such as MoverScore (Zhao et al., 2019), BERTScore (Zhang et al., 2020b) and MARS (Liu et al., 2021b)). 2) Reference-free metrics directly evaluate the quality of generated texts without references. Since unsupervised metrics like perplexity (Brown et al., 1992) and distinct n-grams (Li et al., 2016) can only provide a task-agnostic result which correlates weakly with human judgments (Hashimoto et al., 2019; Tevet and Berant, 2021), most of the reference-free metrics resort to supervised models. Specifically, they are trained to fit human-annotated ratings / labels (such as discriminator scores (Shen et al., 2017)) or distinguish human-written texts from negative samples (such as UNION (Guan and Huang, 2020)). 3) Hybrid metrics contain both referenced and reference-free scores, such as RUBER (Tao et al., 2018; Ghazarian et al., 2019), BLEURT (Sellam et al., 2020) and BARTScore (Yuan et al., 2021).

Compared with existing reference-free metrics which are unsupervised, our metric can support the evaluation of generated texts from different aspects via the full utilization of pre-trained models and the formulation of text infilling tasks, which fits the evaluation protocol of controlled text generation well. Also, in contrast with supervised reference-free metrics, our metric can avoid over-
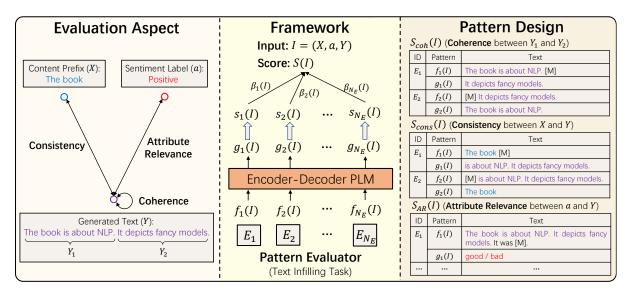
Figure 1: Overview of CTRLEval. **Left**: The three evaluation aspects measure the relationship among content prefixes, attribute labels, and generated texts. **Medium**: The evaluation result $S(I)$ is computed based on the ensemble of the scores from pattern evaluators $E_j (1 \leq j \leq N_E)$. The score $s_j(I)$ of each pattern evaluator E_j is obtained by the generation probability of the encoder-decoder pre-trained language model in the text infilling task, with the input of $f_j(I)$ and the output of $g_j(I)$. **Right**: The evaluation results for three aspects $S_{coh}(I)$ / $S_{cons}(I)$ / $S_{AR}(I)$ are acquired by the corresponding pattern evaluators, respectively.

fitting task-specific data and maintain better generalization ability to evaluate generated texts from different models and with different qualities.

3 Method

3.1 Task Definition and Method Overview

Given the input $I = (X, a, Y)$ which consists of a content prefix X, an attribute label a, and a generated text Y, our goal is to acquire three evaluation results for coherence, consistency and attribute relevance, respectively.

As shown in Figure 1, our main idea is to formulate each evaluation aspect into multiple text infilling tasks and utilize the ensemble of the scores from each task as the final evaluation results. We denote each text infilling task as a *pattern evaluator*, which means evaluation with different input and output patterns. Inspired by the recent works on pattern-exploiting training (Schick and Schütze, 2021a,b) and prompt tuning (Gu et al., 2021), we define each pattern evaluator as $E = (f, g)$, which consists of two pattern functions to build the input and output sequence of text infilling tasks, respectively. The score of each pattern evaluator is acquired from the generation probability of the encoder-decoder pre-trained language model whose pre-training task is to generate the masked part from the remaining texts of the input. For each aspect, we devise multiple pattern evaluators to

alleviate the potential bias caused by the pattern design (Zhao et al., 2021), and weight the scores of all the evaluators to obtain the final result:

$$S(I) = \sum_{j=1}^{N_E} \beta_j(I) \cdot s_j(I) \quad (1)$$

where N_E is the number of pattern evaluators, $S(I)$ denotes the overall score for each aspect, $\beta_j(I)$ is a factor to weight the pattern evaluators of the corresponding aspect and $s_j(I)$ indicates the score of each pattern evaluator based on the generation probability of the pre-trained model.

3.2 Evaluation Aspect

3.2.1 Coherence

Coherence aims to measure whether the sentences in the generated text are semantically relevant to compose a coherent body (Vakulenko et al., 2018; Yuan et al., 2021), which reflects the quality of the generated text itself. Assume that the generated text Y consists of M sentences, i.e., $Y = (Y_1, Y_2, \cdots, Y_M)$, we devise M pattern evaluators $E_j = (f_j, g_j)(1 \leq j \leq M)$ to measure the relevance between each sentence and all the remaining sentences:

$$f_j(I) = Y_{\backslash j} = Y_1 \cdots Y_{j-1} [\text{M}] Y_{j+1} \cdots Y_M \quad (2)$$
$$g_j(I) = Y_j \quad (3)$$

where $Y_{\backslash j}$ indicates the generated text Y with the j-th sentence replaced by a mask token $[\text{M}]$. The score of each pattern evaluator E_j can be computed via the log probability of the pre-trained model P_θ:

$$s_j(I) = \log P_\theta(g_j(I)|f_j(I)) = \log P_\theta(Y_j|Y_{\backslash j}) \quad (4)$$

Since specific and informative sentences are more likely to impact the quality of the whole text, we adopt normalized inverse sentence frequency (NISF) (Zhang et al., 2018) of the output sentence which can reflect its specificity to weight each pattern evaluator:

$$\beta_j(I) = \text{NISF}(Y_j) = \frac{\text{ISF}(Y_j)}{\sum_{k=1}^M \text{ISF}(Y_k)} \quad (5)$$

$$\text{ISF}(Y_j) = \max_{w \in Y_j} \text{IWF}(w) \quad (6)$$

where the inverse sentence frequency (ISF) of Y_j is computed by the maximum inverse word frequency (IWF) of the words in Y_j. We estimate IWF on a general corpus BookCorpus (Zhu et al., 2015), which is commonly adopted as the pre-training dataset in the existing works (Devlin et al., 2019):

$$\text{IWF}(w) = \frac{\log(1 + |C|)}{f_w} \quad (7)$$

where $|C|$ indicates the total number of sentences in BookCorpus and f_w denotes the number of sentences containing the word w. Thus, the evaluation result of coherence can be obtained by the ensemble of the scores from all the pattern evaluators:

$$S_{coh}(I) = \sum_{j=1}^M \text{NISF}(Y_j) \cdot \log P_\theta(Y_j|Y_{\backslash j}) \quad (8)$$

3.2.2 Consistency

Consistency aims to evaluate whether the generated text is consistent to the content prefix (Celikyilmaz et al., 2020; Rashkin et al., 2020). We devise two symmetric pattern evaluators $E_{X \to Y}$ and $E_{Y \to X}$ to evaluate the consistency between the content prefix and the generated text as follows:

$$f_{X \to Y}(I) = X\,[\text{M}], g_{X \to Y}(I) = Y_{\backslash X} \quad (9)$$

$$f_{Y \to X}(I) = [\text{M}]\,Y_{\backslash X}, g_{Y \to X}(I) = X \quad (10)$$

where $Y_{\backslash X}$ denotes the remaining part of the generated text without the prefix. Similar to coherence, we still adopt the log probability of the pre-trained model as the pattern evaluator's score and weight

them with normalized inverse sentence frequency to obtain the final result of consistency:

$$S_{cons}(I) = \text{NISF}(Y_{\backslash X}) \cdot \log P_\theta(Y_{\backslash X}|X\,[\text{M}])$$
$$+ \text{NISF}(X) \cdot \log P_\theta(X|[\text{M}]\,Y_{\backslash X}) \quad (11)$$

3.2.3 Attribute Relevance

Attribute relevance aims to measure whether the generated text satisfies the attribute label (Dathathri et al., 2020). To probe the relevance between generated texts and attribute labels, we first introduce a verbalizer $v(\cdot)$ which maps all the attribute labels a in the attribute set $\mathcal{A}$ to the corresponding words (Schick and Schütze, 2021a). Then, we design the pattern evaluators $E_j = (f_j, g_j)(1 \le j \le N_{AR})$ where $f_j(\cdot)$ adds prompts and a mask token to the generated text, and $g_j(\cdot)$ is set to be a verbalizer:

$$f_j(I) = \text{Concat}(\text{Prompt}_j, [\text{M}], Y) \quad (12)$$

$$g_j(I) = v_j(a) \quad (13)$$

where $\text{Concat}(\cdot)$ indicates the concatenation of the prompt, the mask token, and the generated text in some order. We give an example for the pattern design of attribute relevance which is also shown in Figure 1. In this example, the attribute is set to be the sentiment $\mathcal{A} = \{\text{Positive}, \text{Negative}\}$, while the patterns are designed as $f(I) = $ "Y It was $[\text{M}]$." and $g(I) = v(\text{Positive/Negative}) = \text{good/bad}$.

Inspired by the existing works (Schick and Schütze, 2021a), we use the generation probability of the corresponding label word over all the label words as the score of the pattern evaluator:

$$s_j(I) = \frac{P_\theta(v_j(a)|f_j(I))}{\sum_{a' \in \mathcal{A}} P_\theta(v_j(a')|f_j(I))} \quad (14)$$

Based on the assumption that the pattern evaluator is adequate to measure the data sample if the words of all the attribute labels are easily generated, we devise the unnormalized weighted score of each evaluator as the sum of generation probabilities over all the attribute labels:

$$w_j(I) = \sum_{a' \in \mathcal{A}} P_\theta(v_j(a')|f_j(I)) \quad (15)$$

$$\beta_j(I) = \frac{w_j(I)}{\sum_{k=1}^{N_{AR}} w_k(I)} \quad (16)$$

Similarly, the evaluation result of attribute relevance can be acquired by the weighted sum of all the pattern evaluators' scores:

$$S_{AR}(I) = \sum_{j=1}^{N_{AR}} \beta_j(I) \cdot s_j(I) \quad (17)$$

Task	#Prefixes	#Labels	#Models	#Samples	#Ratings (per sample)	Length	Krippendorff's α
Sentiment	15	2	4	360	5	54.2	0.626
Topic	20	4	4	960	5	55.7	0.622

Table 1: Statistics of the evaluation set, including the number of the prefixes / attribute labels / generation models / samples / ratings (per sample), the average length of each sample and Krippendorff's α.

4 Experiment

4.1 Datasets

Since there is no standard benchmark dataset for evaluating controlled text generation, we construct an evaluation set to measure the correlation between automatic metrics and human judgments.

Task: We choose sentiment-controlled and topic-controlled text generation as the benchmark tasks, which are widely used in the existing works (Dathathri et al., 2020; Chan et al., 2021). These two tasks require the models to generate texts conditioned on the given prefixes and sentiment / topic labels, respectively. In the task of sentiment-controlled text generation, we follow PPLM (Dathathri et al., 2020) and CoCon (Chan et al., 2021) to adopt 15 prefixes and 2 sentiment labels (i.e., positive and negative). As for topic-controlled text generation, we follow CoCon (Chan et al., 2021) to adopt 20 prefixes and 4 topic labels (i.e., computers, politics, religion, and science).

Generation Models: We consider various generation models including CTRL (Keskar et al., 2019), PPLM (Dathathri et al., 2020), GeDi (Krause et al., 2020), and CoCon (Chan et al., 2021). These representative models support both the sentiment-controlled and topic-controlled text generation tasks, and cover different levels of generation abilities. We make these models generate 3 different samples for each unique pair of prefixes and attribute labels. We set the maximum length of generated texts to be 80 and remove the last sentence if it is not complete. We directly use the generation results if they have been released by the original papers. Otherwise, we run the original codes to obtain the generation results.

Human Annotation: We collect human ratings on the generated texts from Amazon Mechanical Turk (AMT). Each survey of AMT contains a prefix, an attribute label, and five generated texts including (a) four generated texts from the above four models respectively, and (b) one negative sample which is constructed by perturbing (e.g. sentence shuffling and dropping) another sample from the evaluation set (Guan et al., 2021). We ask annotators to rate

Task	#Seed Prompts	#Prompts	#Verbalizers	#Evaluators
Sentiment	3	24	3	72
Topic	4	32	1	32

Table 2: Statistics of the pattern evaluators in attribute relevance. The number of evaluators is obtained by multiplying the number of prompts and verbalizers.

these texts with a 1-5 Likert scale for each aspect. To control the annotation quality, we discard the submissions if the annotator assigns a higher rating to the negative sample than other texts. We ensure that each generated text contains 5 valid ratings for each aspect, where the average value of valid ratings is used as the human judgments. We also calculate Krippendorff's α (Krippendorff, 2018) to show the agreement of human ratings, which is 0.626 / 0.622 for sentiment-controlled / topic-controlled text generation tasks, respectively.

The statistics of the evaluation set are shown in Table 1.

4.2 Implementation Details

We choose PEGASUS (Zhang et al., 2020a) as our base model in the overall result and also explore other pre-trained models in §4.8. The hyper-parameters of Transformer blocks are the same as PEGASUS-large with 568M parameters. As for the pattern evaluators in attribute relevance involving prompts and verbalizers which need to be additionally designed, we follow BARTScore (Yuan et al., 2021) to first adopt manually devised seed prompts and verbalizers in the existing works (Schick and Schütze, 2021a,b), and then collect paraphrases to automatically expand our evaluator set. The statistics of pattern evaluators in attribute relevance are presented in Table 2. More details about the specific design of prompts and verbalizers are included in Appendix A.

4.3 Baselines

We choose several state-of-the-art reference-free metrics as our baselines:

Perplexity (PPL) (Brown et al., 1992): This method calculates the perplexity of generated texts

Task	Sentiment						Topic					
Aspect	Coherence			Consistency			Coherence			Consistency		
Metric	r	ρ	τ	r	ρ	τ	r	ρ	τ	r	ρ	τ
DisScore	0.2938	0.2329	0.1664	0.2010	0.1662	0.1178	0.1526	0.1315	0.0937	0.0053	0.0072	0.0051
UNION	0.2317	0.2571	0.1836	0.1925	0.1422	0.1009	0.1628	0.1300	0.0924	0.0664	0.0777	0.0553
BLEURT	0.2585	0.2606	0.1850	0.2382	0.2012	0.1445	0.1631	0.1428	0.1016	0.0433	0.0607	0.0443
PPL-GPT	0.3376	0.3310	0.2350	0.1881	0.1672	0.1203	0.1459	0.1316	0.0940	0.1013	0.0841	0.0595
PPL-PEGASUS	0.3901	0.3860	0.2743	0.2728	0.2513	0.1808	0.1420	0.1313	0.0929	0.1883	0.1771	0.1235
BARTScore	0.3880	0.3848	0.2736	0.2682	0.2533	0.1804	0.1599	0.1325	0.0939	0.1528	0.1408	0.0978
BARTScore-PEGASUS	0.3853	0.3712	0.2653	0.2480	0.2267	0.1630	0.1638	0.1493	0.1048	0.1539	0.1362	0.0953
CTRLEval (Ours)	**0.4395**	**0.4208**	**0.3044**	**0.3226**	**0.3096**	**0.2235**	**0.2403**	**0.2245**	**0.1582**	**0.2342**	**0.2281**	**0.1595**

Table 3: Pearson (r), Spearman (ρ), and Kendall (τ) correlations of coherence and consistency in sentiment-controlled and topic-controlled text generation.

Task	Sentiment			Topic		
Aspect	Attr. Rel.			Attr. Rel.		
Metric	r	ρ	τ	r	ρ	τ
DisScore	0.2213	0.2914	0.2068	0.3624	0.2777	0.1969
UNION	-0.0133	-0.0324	-0.0219	-0.0483	-0.0635	-0.0455
BLEURT	0.0801	0.0652	0.0467	0.1040	0.0841	0.0604
PPL-GPT	-0.0197	-0.0472	-0.0338	0.0853	0.1084	0.0769
PPL-PEGASUS	0.0356	-0.0070	-0.0083	0.0611	0.0662	0.0480
BARTScore	-0.0006	-0.0488	-0.0372	0.0776	0.0853	0.0603
BARTScore-PEGASUS	0.0336	-0.0271	-0.0221	0.0605	0.0567	0.0402
CTRLEval (Ours)	**0.2861**	**0.3008**	**0.2111**	**0.5189**	**0.4006**	**0.2865**

Table 4: Pearson (r), Spearman (ρ), and Kendall (τ) correlations of attribute relevance in sentiment-controlled and topic-controlled text generation. Note that the baselines which are not trained on attribute-annotated corpora can hardly measure the relevance between generated texts and attribute labels, thereby causing low correlations.

with a language model. We use GPT (Radford et al., 2018) and PEGASUS (Zhang et al., 2020a) as the base models since GPT is commonly used in the existing works (Dathathri et al., 2020) and PEGASUS is our base model. They are denoted as **PPL-GPT** and **PPL-PEGASUS**, respectively.

Discriminator Score (DisScore) (Kannan and Vinyals, 2017; Chan et al., 2021): This method trains a discriminator with different objectives. We adopt the IMDB movie review dataset (Maas et al., 2011) / HuffPost News category dataset[2] (Misra, 2018) for sentiment-controlled / topic-controlled text generation tasks, respectively. For coherence and consistency, the discriminator is trained to distinguish human-written texts from manually constructed negative samples, where the ratio of positive and negative samples is 1:1. For attribute

relevance, it is trained based on the sentiment / topic classification task, respectively (Chan et al., 2021). Both the sentiment and topic discriminators are implemented based on BERT (Devlin et al., 2019) and they achieve 94.15% / 91.54% on the corresponding test set, respectively.

UNION (Guan and Huang, 2020): This method is a self-supervised metric which is trained to distinguish human-written texts from the automatically perturbed negative samples with well-designed negative sampling strategies and multi-task learning. We use the same datasets as the discriminator score to train UNION.

BLEURT (Sellam et al., 2020): This method is a supervised metric which is pre-trained on synthetic examples and then fine-tuned to fit human ratings. We used the same instruction in §4.1 to additionally annotate the generated texts to construct the training set for BLEURT, whose amount is the same as the evaluation set. There is no overlap between BLEURT's training set and the evaluation set.

BARTScore (Yuan et al., 2021): This method utilizes the generation probabilities of BART (Lewis et al., 2020) to measure the relationship among sources, hypotheses, and references. Since this metric simultaneously contains referenced and reference-free parts, we only use the reference-free score in our experiments. We also use PEGASUS (Zhang et al., 2020a) as the base model for a fair comparison, which is denoted as **BARTScore-PEGASUS**.

4.4 Overall Result

We follow the existing work (Guan and Huang, 2020; Yuan et al., 2021) to adopt Pearson (r), Spearman (ρ), and Kendall (τ) correlation coefficients between automatic metrics and human judgments

to measure the performance of different metrics.

The overall results on sentiment-controlled and topic-controlled text generation are shown in Table 3 and 4. We can observe that CTRLEval outperforms other baselines with a large margin, indicating the effectiveness of our metric on different evaluation aspects. In Table 4, unsupervised baselines can hardly measure the relevance between generated texts and attribute labels because they only provide a task-agnostic score which is weakly relevant to this specific aspect. For comparison, our metric, which supports the evaluation for different aspects of generated texts via the design of text infilling tasks, can obtain much better performance and even outperform the supervised baselines.

4.5 Ablation Study

Metric	Aspect		
	Coherence	Consistency	Attr. Rel.
CTRLEval (Ours)	**0.2403**	**0.2342**	**0.5189**
Weight of Pattern Evaluators (w/o β)			
w/ mean($\cdot$)	0.2295	0.1927	0.5091
w/ max($\cdot$)	0.2323	0.1772	0.5170
w/ min($\cdot$)	0.1518	0.1559	0.4153
Pattern Function (w/o $f\&g$)			
w/ PPL-GPT-PF	0.2041	0.2169	0.4376
w/ BARTScore-PF	0.1236	0.1843	0.3972

Table 5: Pearson correlation of ablation models in topic-controlled text generation.

To further investigate the effect of each module, we conduct ablation studies on the weight of pattern evaluators and the design of pattern functions. For the weight of evaluators, we use the mean, maximum and minimum values of all the evaluators as the final result rather than the weighted sum based on the factor β. As for the design of pattern functions, we fix the base model and replace our input and output patterns ($f\&g$) with those of PPL-GPT (Radford et al., 2018) and BARTScore (Yuan et al., 2021). The pattern functions of these ablation models are not designed for text infilling tasks. Both of them remove the mask token in the input pattern, and PPL-GPT additionally places the input pattern at the beginning of the output pattern.

The results in Table 5 show that each module in our metric contributes to the final performance. As for the weight of evaluators, we can observe that our weight factor performs better than common aggregation functions especially in consistency, indi-

cating the necessity of the well-designed ensemble method when the number of pattern evaluators is small. Also, our pattern functions outperform those of other baselines, thereby showing the effectiveness of text infilling tasks which can fully utilize pre-trained models in an unsupervised setting.

4.6 Analysis on Generalization Ability

Generalization ability is essential for automatic metrics to evaluate open-ended text generation models. In this section, we will test whether our metric can be generalizable to measure the generated texts faced with model drift and quality drift.
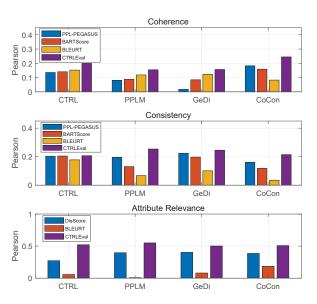
4.6.1 Model Drift

Figure 2: Pearson correlation on the generated results from four generation models in the task of topic-controlled text generation.

To measure whether CTRLEval is reliable to assess the generated results of different models, we split the evaluation set into four subsets based on the generation model and calculate Pearson correlation between each metric and human judgments.

The results in Figure 2 show that our metric can outperform other baselines on the generated texts of all the generation models. Simultaneously, CTRLEval can achieve stable performance with smaller variances when evaluating different generation models, indicating that our metric can generalize to the model drift better.

4.6.2 Quality Drift

To evaluate the generalization ability of CTRLEval on the generated texts with different qualities, we follow the existing work (Sellam et al., 2020; Guan
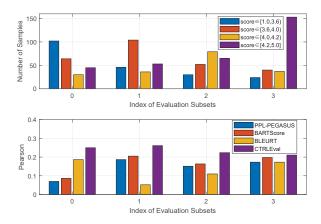
Figure 3: **Top**: The number of samples with different coherence scores in the four biased evaluation subsets. **Bottom**: Pearson correlation of different metrics on the biased evaluation subsets.

Figure 4: Pearson correlation of the models with different numbers of evaluators.

and Huang, 2020) to construct four biased subsets based on the coherence score of topic-controlled text generation. We first sort all the samples in the evaluation set and use the quartiles to split them into four subsets with the index from 0 to 3. Then, we create four biased subsets. For the jth subset, we sampled the generated texts which belong to the original i^{th} subset with a probability of $\frac{1}{|j-i|+1}$ where $i, j = 0, 1, 2, 3$. Thus, the four biased subsets have different distributions of generated texts with different qualities, as shown in Figure 3.

We then calculate the Pearson correlation between each metric and human judgments. The results in Figure 3 show that CTRLEval has higher correlations than the baselines on the evaluation subsets with different qualities. Also, our metric can achieve more stable performance on different subsets, which shows our better generalization ability to deal with quality drift.

4.7 Analysis on the Number of Evaluators

To investigate how the number of pattern evaluators affects the performance, we randomly sample the evaluators 20 times when evaluating attribute relevance in topic-controlled text generation, and illustrate mean values and standard deviations of each number of evaluators in Figure 4.

Figure 4 shows that as the number of evaluators increases, the mean value of our performance can be persistently improved while the standard deviation is gradually reduced. This demonstrates the necessity of devising multiple pattern evaluators for each aspect, which can alleviate the bias brought by the pattern design. The comparison between the pattern functions of CTRLEval and other base-
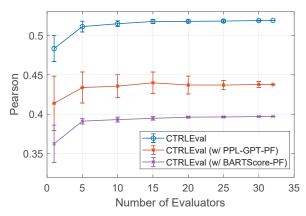
lines indicates our superior performance on all the numbers of evaluators.

4.8 Analysis on Base Model

Base Model	#Param	Aspect		
		Coherence	Consistency	Attr. Rel.
PEGASUS	568M	0.3044	0.2235	**0.2111**
BART	400M	**0.3123**	0.1650	0.1951
T5	770M	0.2930	**0.2350**	0.2075

Table 6: Kendall correlation of CTRLEval with different base models in sentiment-controlled text generation. #Param means the number of parameters.

Since our method can adapt to different pretrained models whose pre-training task is text infilling, we additionally choose BART (Lewis et al., 2020) and T5 (Raffel et al., 2020) as our base model, and present the results in Table 6.

Table 6 shows that PEGASUS and T5 obtain comparable performance on all the evaluation aspects, which indicates that our well-designed text infilling tasks can be transferable to T5 without considerable modification. As for BART which performs worse on consistency and attribute relevance, we conjecture that the fewer parameters and the form of pre-training tasks may limit the performance. Since the pre-training task of BART is to generate the complete text rather than only the masked part of the input text, it may not be good at the evaluation involving a short span of texts, such as the prefix in the evaluation of consistency and the label word in attribute relevance.

We also provide the analysis on the number of parameters in Appendix B and the case study in Appendix C.

5 Discussion

Extension to More Control Variables: In this paper, we evaluate the relationship between generated texts and two control variables (including content prefixes and attribute labels) via consistency and attribute relevance, respectively. We can also extend our metric to other control variables by designing additional pattern evaluators to measure the relationship between generated texts and each variable, respectively. We will further investigate the extensibility of our metric in the future work.

Design of Pattern Evaluators: With the rapid development of prompt tuning, recent works have proposed new methods on the design of prompts and verbalizers (Gao et al., 2021; Lester et al., 2021), which provide alternatives to our metric in attribute relevance. Also, the weight factor of each evaluator can be set as diversity metrics (Hashimoto et al., 2019) besides NISF in coherence and consistency. We will leave the exploration of more settings on pattern evaluators as the future work.

6 Conclusion

We present an unsupervised reference-free metric called CTRLEval for evaluating controlled text generation. This metric formulates the evaluation of different aspects into multiple text infilling tasks, and utilizes the ensemble of generation probabilities from a pre-trained model in different tasks as the evaluation result. Experimental results indicate that CTRLEval obtains higher correlations with human judgments and shows better generalization ability for addressing model drift and quality drift.

Acknowledgments

Ethics Statement

We construct an evaluation set for evaluating controlled text generation. The data samples in this set are all from the existing works with open-source codes, model checkpoints, and generated results. We directly use the generated results if the authors have released them. Otherwise, we adopt the same setting as the original papers to make these models generate texts. We do not apply extra selection strategies to the generated results.

We resort to Amazon Mechanical Turk (AMT) for the annotation of this evaluation set. We do not invade the privacy or collect personal information of annotators. We pay each annotator $0.06 for each survey including four generated texts and one negative sample. The payment is determined based on the length of data samples. We additionally ask annotators to check whether there is a potential ethical problem in the data, and remove these problematic data in the evaluation set. After annotation on AMT, we manually review all the annotated samples from an ethical perspective. However, we admit that there may still exist unpredictable bias in this evaluation set.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: an automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Peter F. Brown, Stephen Della Pietra, Vincent J. Della Pietra, Jennifer C. Lai, and Robert L. Mercer. 1992. An estimate of an upper bound for the entropy of english. *Comput. Linguistics*, 18(1):31–40.

Asli Celikyilmaz, Elizabeth Clark, and Jianfeng Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.

Alvin Chan, Yew-Soon Ong, Bill Pung, Aston Zhang, and Jie Fu. 2021. Cocon: A self-supervised approach for controlled text generation. In *9th International Conference on Learning Representations*.

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 *8th International Conference on Learning Representations*.

Mingkai Deng, Bowen Tan, Zhengzhong Liu, Eric Xing, and Zhiting Hu. 2021. Compression, transduction, and creation: A unified framework for evaluating natural language generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7580–7605.

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, pages 4171–4186.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. In *Proceedings of the Workshop on Stylistic Variation*, pages 94–104.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2021. Making pre-trained language models better few-shot learners. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 3816–3830.

Cristina Garbacea, Samuel Carton, Shiyan Yan, and Qiaozhu Mei. 2019. Judge the judges: A large-scale evaluation study of neural language models for online review generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3966–3979.

Sarik Ghazarian, Johnny Wei, Aram Galstyan, and Nanyun Peng. 2019. Better automatic evaluation of open-domain dialogue systems with contextualized embeddings. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 82–89.

Yuxian Gu, Xu Han, Zhiyuan Liu, and Minlie Huang. 2021. Ppt: Pre-trained prompt tuning for few-shot learning. *arXiv preprint arXiv:2109.04332*.

Jian Guan and Minlie Huang. 2020. UNION: an unreferenced metric for evaluating open-ended story generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 9157–9166.

Jian Guan, Zhexin Zhang, Zhuoer Feng, Zitao Liu, Wenbiao Ding, Xiaoxi Mao, Changjie Fan, and Minlie Huang. 2021. Openmeva: A benchmark for evaluating open-ended story generation metrics. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 6394–6407.

Tatsunori B. Hashimoto, Hugh Zhang, and Percy Liang. 2019. Unifying human and statistical evaluation for natural language generation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1689–1701.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017. Toward controlled generation of text. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pages 1587–1596.

Anjuli Kannan and Oriol Vinyals. 2017. Adversarial evaluation of dialogue models. *arXiv preprint arXiv:1701.08198*.

Pei Ke, Jian Guan, Minlie Huang, and Xiaoyan Zhu. 2018. Generating informative responses with controlled sentence function. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1499–1508.

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. 2020. Gedi: Generative discriminator guided sequence generation. *arXiv preprint arXiv:2009.06367*.

Klaus Krippendorff. 2018. *Content analysis: An introduction to its methodology*. Sage publications.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pretraining 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 *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021a. 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*, pages 6691–6706.

Ruibo Liu, Jason Wei, and Soroush Vosoughi. 2021b. Language model augmented relevance score. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing*, pages 6677–6690.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *The 49th Annual Meeting of the Association for*

Computational Linguistics: Human Language Technologies, pages 142–150.

Rishabh Misra. 2018. News category dataset.

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.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 4274–4295.

Timo Schick and Hinrich Schütze. 2021a. Exploiting cloze-questions for few-shot text classification and natural language inference. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 255–269.

Timo Schick and Hinrich Schütze. 2021b. Few-shot text generation with natural language instructions. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 390–402.

Thibault Sellam, Dipanjan Das, and Ankur P. Parikh. 2020. BLEURT: learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892.

Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi S. Jaakkola. 2017. Style transfer from non-parallel text by cross-alignment. In *Advances in Neural Information Processing Systems*, pages 6830–6841.

Chongyang Tao, Lili Mou, Dongyan Zhao, and Rui Yan. 2018. RUBER: an unsupervised method for automatic evaluation of open-domain dialog systems. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 722–729.

Guy Tevet and Jonathan Berant. 2021. Evaluating the evaluation of diversity in natural language generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 326–346.

Svitlana Vakulenko, Maarten de Rijke, Michael Cochez, Vadim Savenkov, and Axel Polleres. 2018. Measuring semantic coherence of a conversation. In *The Semantic Web - ISWC 2018 - 17th International Semantic Web Conference*, volume 11136, pages 634–651.

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.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2021. Bartscore: Evaluating generated text as text generation. *Advances in Neural Information Processing Systems*, 34.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020a. PEGASUS: pre-training with extracted gap-sentences for abstractive summarization. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119, pages 11328–11339.

Ruqing Zhang, Jiafeng Guo, Yixing Fan, Yanyan Lan, Jun Xu, and Xueqi Cheng. 2018. Learning to control the specificity in neural response generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1108–1117.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020b. Bertscore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 563–578.

Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pages 12697–12706.

Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. 2018. Emotional chatting machine: Emotional conversation generation with internal and external memory. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, pages 730–739.

Xianda Zhou and William Yang Wang. 2018. Mojitalk: Generating emotional responses at scale. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*, pages 1128–1137.

Yukun Zhu, Ryan Kiros, Richard S. Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *IEEE International Conference on Computer Vision*, pages 19–27.

A   Pattern Evaluator for Attribute Relevance

We first choose the prompts and verbalizers which have been shown to work well in the existing works on few-shot text classification (Schick and Schütze, 2021a; Gao et al., 2021) and generation (Schick and Schütze, 2021b) as the seed prompts and verbalizers. Then, we expand our prompt set with the following rules: 1) Switching the order of generated texts, prompts, and mask tokens; 2) Collecting the paraphrases of seed prompts just as BARTScore (Yuan et al., 2021) does. All the prompts and verbalizers which are used in our experiments are shown in Table 8.

B   Analysis on the Number of Parameters

Base Model	#Param	Aspect		
		Coherence	Consistency	Attr. Rel.
T5-small	60M	0.2389	0.1495	0.1765
T5-base	220M	0.2847	0.2053	0.1867
T5-large	770M	**0.2930**	**0.2350**	**0.2075**

Table 7: Kendall correlation of CTRLEval with T5-small, T5-base, and T5-large in sentiment-controlled text generation. #Param means the number of parameters.

We further conduct experiments on the base model with different numbers of parameters. Since the authors of PEGASUS (Zhang et al., 2020a) do not release the model checkpoint of PEGASUS-base, we choose T5-small, T5-base and T5-large (Raffel et al., 2020) as our base models respectively, and present the results in Table 7. The results show that larger numbers of parameters can benefit the model performance while degrading the computation efficiency.

C   Case Study

To intuitively show how our metric works in the evaluation of controlled text generation, we provide some cases on the three evaluation aspects, including coherence (Figure 6), consistency (Figure 6), and attribute relevance (Figure 7). Since the range of various metrics is always different, it

Figure 5: The score and weight of each evaluator for evaluating attribute relevance of the second sample in Figure 7.

may be less meaningful to directly compare the absolute value of each metric. Thus, we follow the existing works (Guan and Huang, 2020; Liu et al., 2021b) to conduct a pairwise comparison on different samples.

The results in Figure 6 and 7 show that our metric can give accordant preferences with human judgments, indicating the effectiveness of our metric on all three evaluation aspects. To further show how each pattern evaluator works in the overall evaluation result, we take the second sample in Figure 7 as an example and visualize the weight $\beta(I)$ and score $s(I)$ in Figure 5. We can observe that most of the pattern evaluators assign high scores to this sample which agree with the human judgment. Simultaneously, the weight factor automatically reduces the effect of low-quality evaluators which also plays an important role in the final evaluation result.

Task	Sentiment	
	Seed Prompt	Expanded Prompt
$f(I)$	Y In summary, it was [M].	In summary, it was [M]. Y Y To sum up, it was [M]. To sum up, it was [M]. Y Y All in all, it was [M]. All in all, it was [M]. Y Y In brief, it was [M]. In brief, it was [M]. Y
	Y It was [M].	It was [M]. Y Y It seems [M]. It seems [M]. Y Y It appears [M]. It appears [M]. Y Y It becomes [M]. It becomes [M]. Y
	Y Really [M]!	Really [M]! Y Y Just [M]! Just [M]! Y Y Actually [M]! Actually [M]! Y Y So [M]! So [M]! Y
$g(I)$	Verbalizer	
	v(Positive, Negative) = {(good, bad), (positive, negative), (great, terrible)}	

Task	Topic	
	Seed Prompt	Expanded Prompt
$f(I)$	Y News: [M]	News: [M] Y Y Article: [M] Article: [M] Y Y Summary: [M] Summary: [M] Y Y Report: [M] Report: [M] Y
	Y It was about [M].	It was about [M]. Y Y It was around [M]. It was around [M]. Y Y It was related to [M]. It was related to [M]. Y Y It was towards [M]. It was towards [M]. Y
	Y It was a piece of [M] news.	It was a piece of [M] news. Y Y It was a [M] article. It was a [M] article. Y Y It was a [M] summary. It was a [M] summary. Y Y It was a [M] report. It was a [M] report. Y
	Y What [M] news!	What [M] news! Y Y What a [M] article! What a [M] article! Y Y What a [M] summary! What a [M] summary! Y Y What a [M] report! What a [M] report! Y
$g(I)$	Verbalizer	
	v(Computers, Politics, Religion, Science) = {(computers, politics, religion, science)}	

Table 8: Prompts and verbalizers used in the evaluation of attribute relevance, where $I = (X, a, Y)$ indicates the prefix, the attribute label, and the generated text, respectively.

Aspect	Coherence				
ID	Generated Text (Y)	PPL-PEGASUS ($\downarrow$)	BARTScore ($\uparrow$)	CTRLEval ($\uparrow$)	Human Rating
1	(1) The book is an excellent introduction to the subject of macroeconomics. (2) It provides a great overview and introduces you to many important concepts such as inflation, interest rates, capital flows, etc. (3) This book should be read by anyone interested in economics or financial markets. (4) I would recommend it for both students who are just starting out with their studies on finance and also seasoned professionals looking for some new perspectives on finance.	2.6089 (29)	-3.2914 (84)	**-1.4495** (3)	**4.8** (55)
2	(1) The book is a wonderful, beautifully designed and beautiful book, and a must-read for anyone interested in the history of the American revolution. (2) It is an excellent introduction to the subject, and has been highly recommended to all of the students I have taught. (3) The author is also a superb speaker, and his lectures are extremely engaging.	**2.6052** (28)	**-2.9722** (22)	-2.1080 (42)	4.6 (127)
Aspect	Consistency					
---	---	---	---	---	---	---
ID	Prefix (X)	Generated Text (Y)	PPL-PEGASUS ($\downarrow$)	BARTScore ($\uparrow$)	CTRLEval ($\uparrow$)	Human Rating
1	The painting	**The painting** is very nice and I am happy with my purchase. It looks great in our living room as a decorative piece rather than an actual picture of the painting. However, it was not packaged well at all. When we received it, there were several pieces broken off from each other on one side so that when you hang it up, they are all hanging crooked.	3.3785 (181)	-3.7656 (185)	**-3.2493** (123)	**3.8** (38)
2	The painting	**The painting** is way too short! But, that's all I can do in this short bit of a post. But for those who don't know I'm a high school art teacher, and I like to teach art, craft, or anything creative to kids at all ages. So I've tried to write a little about the process in the first post.	**3.0690** (122)	**-3.5702** (139)	-4.1035 (259)	2.8 (234)

Figure 6: Evaluation cases on coherence and consistency in sentiment-controlled text generation. The result of each metric is provided by the absolute value of the evaluation score and the rank of this score over all the evaluation results of the corresponding metric. To improve readability, we label each sentence with its order in the evaluation of coherence and highlight the prefix in bold when evaluating consistency.

Aspect	Attribute Relevance				
ID	Sentiment Label (a)	Generated Text (Y)	DisScore ($\uparrow$)	CTRLEval ($\uparrow$)	Human Rating
1	Positive	The book is a collection of stories about the lives and experiences of two young girls who are in love with each other. It starts out with one girl named Jenny and ends up with another girl, Jane. They both have their own problems to deal with, but they end up finding that together they can overcome anything life throws at them. I liked this book, because it was very realistic and had me laughing throughout most of it.	**0.9982** (10)	0.9326 (97)	3.8 (96)
2	Positive	The book is a great read. I was so inspired by the stories and characters and the amazing writing by Kaitlyn (the author) that I decided to write a book. The book is an adventure story set during the time period and is a must read for any fantasy reader. The writing is amazing, I highly recommend reading it!	0.9978 (48)	**0.9858** (38)	**4.6** (4)

Figure 7: Evaluation cases on attribute relevance in sentiment-controlled text generation. The result of each metric is provided by the absolute value of the evaluation score and the rank of this score over all the evaluation results of the corresponding metric.