Controlling Text Generation With Fiction-Writing Modes

Wenjie Zhong[1,2], Jason Naradowsky[1], Hiroya Takamura[2], Ichiro Kobayashi[2,3], Yusuke Miyao[1,2]
[1]The University of Tokyo, [2]AIST, [3]Ochanomizu University,
zvengin@is.s.u-tokyo.ac.jp

Abstract

We explore incorporating concepts from writing skills curricula into human-machine collaborative writing scenarios, focusing on adding **writing modes** as a control for text generation models. Using crowd-sourced workers, we annotate a corpus of narrative text paragraphs with writing mode labels. Classifiers trained on this data achieve an average accuracy of ~ 87% on held-out data. We fine-tune a set of large language models to condition on writing mode labels, and show that the generated text is recognized as belonging to the specified mode with high accuracy.

1 Introduction

Large-scale pre-trained language models (PLMs) have demonstrated a remarkable aptitude for generating text with an exceptional degree of fluency and structure [1], sparking renewed efforts to utilize them for the purpose of generating narrative fiction. Recent work has explored various ways of controlling PLMs, using sentiment [2], style [3], and even character information [4], in an attempt to cater the generated text to an author's intentions.

However, the aforementioned controls deal primarily with **static** attributes of text; an attribute like style is more synonymous with an entire author or book than with a single passage of text. Less attention has been paid to designing effective control factors for the real demands of human authors in collaborative writing settings, where authors typically exercise more **dynamic** control over their writing, at the sentence or paragraph level. Here we find inspiration from the creative writing literature, where the notion of a **fiction writing mode** is frequently presented as an important concept to consider when crafting narrative fiction.

A fiction-writing mode (also referred to as a rhetorical mode) is a particular manner of writing, encapsulating the focus, style, and pacing of the text (among other things) [5].

Summary: A boy was walking along a path in the forest, when he came across a heap of leaves.

Mode	Story
Dialogue	"Here are some leaves," he **whispered**. "They were wet when we came, and are wet now. I'll lie them down and wait." "What is it?" I **exclaimed**. "If you will stand still," **said** my boy, "I will show you … "
Action	He **stood** for a moment **looking** at me, then quietly he **picked up** the leaves, and **carrying** them in his hand, **climbed** to the top of the heap, and **examined** them …
Description	This heap consisted of **dead leaves**, many of them **rotten**, and still **wet**, with one or two **lying flat** on the ground, others **lying up against** the branches. The first to fall was the one I had thought **dead**. It had been crushed by the wind. …

Figure 1 Example of expanding the **Summary** into stories using different writing **Mode**s. The bold words imply the particular manner of expression in that mode. **Dialogue** focuses on the utterances spoken by characters, **Action** on the motion of characters, and **Description** on the depiction of characters or places.

Figure 1 illustrates how the same event can be described in different ways depending on the writing mode, using the three most common types, **Dialogue**, **Action**, and **Description**. Skilled authors proficiently use writing modes as a stylistic choice to engage readers and progress the narrative (see Section 2 for more detail). Thus, we expect to provide the fiction-writing mode to users to control the models to generate text with different styles.

To this end, we are faced with a challenge. To the best of our knowledge, there is no available dataset annotated with writing modes to train generation models. We create a Fiction wRIting moDE dataset (**FRIDE** dataset) containing 1,736 fiction paragraphs annotated by crowd-source workers with the three writing mode labels. Subsequently, we train a classifier on the **FRIDE** dataset and use it to anno-

Dia.	Act.	Des.	Unc.	Total	Len.(std)	Kappa
370	385	300	681	1,736	110(52)	0.64

Table 1 The number of instances for dialogue (**Dia.**), action (**Act.**), description (**Des.**), and uncertain (**Unc.**) modes in the dataset. **Kappa** is the inter-annotator agreement and **Len.(std)** is the average token number in each instance and its standard deviation.

tate paragraphs of a large fiction corpus in order to create a larger-scale dataset. Using the established paradigm of training conditional text generation models by summarizing and reconstructing text [6], the dataset is used to train models which can be conditioned on a writing mode label.

Through the automatic evaluation, we show: (1) the use of writing mode labels with conditional text generation models contributes to average 1.4 and 2.0 points improvements on ROUGE-L and BERTScore; (2) the writing modes of generated text are effectively controlled, and are classified as belonging to the target mode in 87.6% of cases.

2 Fiction-Writing Mode

Fiction-writing modes have long been proposed as a useful abstraction in the study of literature and creative writing [7, 5], dating as far back as Aristotle [8]. While there is no consensus on the categorization of writing modes, most sources prefer to introduce at least three modes: (1) **Dialogue**, direct quotation of characters speaking, (2) **Action**, an account of a series of events, one after another, chronologically, and (3) **Description**, a more detailed inspection of people, places, or things and their properties. These are the three major writing modes which are the focus of study in this paper.

Just as there is no agreement on how best to categorize writing modes, there is also no consensus on what text exhibits a particular mode. Even a single sentence can exhibit multiple writing modes, in varying degrees. However, for the purpose of this work, we assume that each paragraph can be categorized as exhibiting a single writing mode.

FRIDE Dataset In order to train models which generate text in a specified writing mode, we must first create a dataset, which we refer to as Fiction-wRIting moDE dataset (**FRIDE** dataset), which pairs paragraphs of narrative text with their corresponding writing mode labels. However, directly annotating writing modes on a large-scale narrative dataset is expensive and time-consuming. We first collect a modestly sized dataset from crowd-sourced workers, and utilize it to train a writing mode classifier. The classifier can then be used to provide high-confidence labels to a much larger dataset of narrative text paragraphs, on a scale suitable for training large text generation models.

Paragraphs for annotation are collected from fiction books sourced from Project Gutenberg[1)] (128 books) and, namely, for more contemporary writing, Smashwords[2)] (150 books). Each book is divided into paragraphs using Chapterize[3)], and paragraphs longer than 200 words are removed. In situations where a continuous dialogue takes place over paragraph boundaries, we group them into a single paragraph. Each paragraph was annotated with one of the three aforementioned writing modes using Amazon Mechanical Turk (AMT). In addition, we add a fourth category, **Uncertain**, to encompass cases where the writing mode is unclear or does not fit well into the three main modes. All annotators were native English speakers, and three annotators were assigned to each paragraph. Paragraphs were assigned the majority label, or marked as *uncertain* in cases where each annotator provided a different label. We continued the annotation process until we had approximately 1,000 instances labeled and balanced across the three main modes (Table 1).

Writing Mode Classifier While it is possible to use the collected data to train a model, the relatively small pool of examples may cause the model to be sensitive to other text characteristics unrelated to the writing mode. To help alleviate this problem, we train a writing mode classifier and employ it to predict writing modes on a larger collection of texts. We experiment with training three separate classifiers, each trained by fine-tuning a different PLM (BERT [9], XLNet [10], or RoBERTa [11]) on the **FRIDE** dataset. We randomly sample 300 instances from each type of writing mode and divide them using a 1000/100/100 train/dev/test split, with an equal number of each label in each split. An evaluation of these models (Table 2) shows that all models perform similarly. The RoBERTa-based model was used as the final writing mode classifier throughout the remainder of this paper.

FRIDE-XL Dataset In order to construct a larger dataset of writing modes suitable for training mode-conditional text generation models, we utilize the classifier trained in the preceding section on a larger set of texts,

1) https://www.gutenberg.org
2) https://www.smashwords.com
3) https://github.com/JonathanReeve/chapterize

	Precision	Recall	F1
BERT-base	85.7	85.0	85.0
XLNet-base	84.7	84.4	84.3
RoBERTa-base	**86.3**	**85.2**	**85.2**

Table 2 The performance of writing mode classifiers on the **FRIDE** dataset.

extending the previous text to 5,946 fiction books from Project Gutenberg. We leverage the writing mode classifier to assign a writing mode label to each paragraph of books and randomly select 362,880 paragraphs. We refer to this dataset as **FRIDE-XL**.

3 Models

We evaluate writing mode as a control factor on three different PLM architectures: BART [12], T5 [13], and GPT2 [14]. All models have been used previously for text generation but differ in ways that may impact their ability to adhere to the conditioning information and the quality of the generated text. For instance, the larger parameter size and contextual window size of GPT2 has made it a common choice for story generation with long text [15, 16, 17], but smaller models like T5 show great controllability [18]. We assess each of these three models, fine-tuning them to reconstruct paragraphs from the **FRIDE-XL** dataset.

For training conditional text generation models, we follow an established paradigm of summarization, conditioning, and reconstruction [6]. First, each paragraph is summarized using an existing summarization model. Here we use the narrative text summarization [19], and decode using beam search with a beam size of 5 as in that work. We then fine-tune a PLM to reconstruct the original paragraph, conditioning on the summary. In this way, the summary acts as a semantic control: the trained model accepts user summaries and attempts to expand upon them to generate a longer paragraph, embellishing missing and less important details in a reasonable way.

Other forms of information can also be added to the summaries to function as additional controls. The conditioning factors provided to models are:

- **Summary**, generated from the paragraph by a pre-trained model.
- **Context**, the preceding paragraph.
- **Length**, the number of tokens in the paragraph divided into ten equally-sized bins.
- **Writing Mode**, the mode assigned to the paragraph by the classifier as described in Sec. 2.

For T5 and BART, the training methodology is straightforward: we concatenate the controlling information and use it as input to the encoder, training the decoder to generate the original paragraph. For GPT2, which has only a decoder, we concatenate the conditions as prompts.

4 Automatic Evaluation

In this section, we study the influence of model inputs (e.g., summaries, length, and writing modes) on the text quality, and assess to what extent the writing modes of text can be controlled, as measured by automatic metrics.

4.1 Baseline Models

In addition to ablations of our proposed models, we compare against three baseline systems:

GPT2 We finetune GPT2 [14] identically to our proposed system, but using only the preceding paragraph and without other inputs.

PPLM As conventional training requires lots of annotated data, an attribute classifier is employed to guide the pretrained language model to generate text with specified attributes [20] . To adapt the PPLM to our task, we train a writing mode classifier as the attribute classifier on the **FRIDE** dataset. As the writing modes of preceding paragraphs would interfere with the classifier, the PPLM does not take the preceding paragraphs as context.

FIST A system [21] which utilizes keywords instead of summaries to sketch the semantic content of the desired stories is proposed. As there is no prompt in our dataset, following their idea, we infer the keywords from the leading context (the preceding paragraphs) and then generate stories conditioning on the context and keywords.

4.2 Results

We evaluate the models along three axes: fluency, similarity, and controllability, using the test set of the **FRIDE-XL** dataset. The results of our automatic evaluation are shown in Table 3.

Fluency We evaluate fluency using perplexity computed by the pre-trained GPT2 model. We find that there is an average 0.8 decrease in perplexity when summaries are added and 1.2 increase when writing modes are added.

Similarity We evaluate the similarity of the generated texts to the target texts using BLEU-4 [22], ROUGE-L [23],

	Model Inputs			Quality				Controllability (Accuracy)		
	L	**S**	**M**	**PPL** ↓	**B4** ↑	**RL** ↑	**BS** ↑	**Dialogue**	**Action**	**Description**
GPT2				24.41	0.95	15.02	45.61	73.33	20.28	21.67
FIST		✓		23.68	0.99	15.41	45.97	85.00	41.11	46.94
PPLM			✓	24.10	1.07	14.50	42.84	93.05	32.22	49.72
GPT2	✓			**19.29**	1.06	15.74	46.33	72.50	22.78	22.50
	✓	✓		18.84	1.14	15.97	46.70	83.33	38.61	45.56
	✓		✓	20.29	1.09	16.00	46.82	97.78	67.78	71.11
	✓	✓	✓	19.89	1.16	16.10	47.28	98.06	75.00	79.72
T5	✓			24.98	1.14	16.22	46.42	67.78	25.28	23.61
	✓	✓		23.80	1.21	16.32	46.78	80.56	47.78	46.67
	✓		✓	26.12	1.16	16.30	47.07	**99.44**	85.00	78.33
	✓	✓	✓	25.06	**1.20**	**16.52**	47.10	98.33	83.61	83.06
BART	✓			23.87	1.07	16.19	46.32	69.44	19.17	18.33
	✓	✓		23.49	1.17	16.33	47.12	86.67	47.78	48.89
	✓		✓	25.44	1.11	16.25	47.30	98.06	82.50	**88.06**
	✓	✓	✓	24.24	**1.20**	16.27	**47.30**	97.78	**85.56**	82.78

Table 3 Automatic evaluation on quality and controllability as model inputs (summaries (**S**), length (**L**), and writing modes (**M**)) vary. Quality is evaluated by perplexity (**PPL**), BLEU-4 **(B4)**, ROUGE-L (**RL**), BERTScore (**BS**), and controllability is measured by the accuracy of the generated stories matching the specified writing mode. controlling the writing modes of stories when the writing modes (**M**) are specified as **Dialogue**, **Action**, and **Description**. The inputs such as summaries (**S**), length (**L**), and writing modes (**M**) for the evaluation of quality and controllability are respectively inferred from the leading context and the target stories.

and BERTScore [24]. We observe a consistent improvement across all models as the amount of conditioning context increases, and that the models using writing mode factors outperform those without.

Controllability Lastly, we evaluate the controllability of mode-controlled models. For each paragraph, a target writing mode is chosen using the writing mode classifier, and used as conditioning for a text generation model. The classifier is then used to predict the writing mode of the generated text, and we measure the accuracy of generating stories with the specified writing modes.

On average, including writing mode as condition improves the accuracy of generating text which is classified as that mode, but the effect varies drastically by the specific mode. For Action and Description modes, the inclusion of writing mode conditioning improves accuracy on average by 45.4% and 45.6%, respectively, compared to none mode-conditioned models. For dialogue, the improvement is 21.5%, relatively lower.

It is interesting to note that the inclusion of summaries to the length-only model results in significant improvements to the controllability of the text. This implies that the pretrained models are able to naturally infer the intended writing mode from the summaries to some degree, with modest accuracy ($\sim$ 44%) on average for Action and Description modes, and up to 86% on Dialogue with BART. Summaries may contain some cues about the intended modes, especially, the summaries for Dialogue have strong cues (**said**, **replied**, **argued**, ...) in most cases. However, the consistently significant increase of accuracy scores when conditioning on writing modes illustrates the effectiveness of modes as a control factor.

5 Conclusion

In this work, we introduced writing modes as a control for human-machine collaborative writing scenarios and showed that training models to condition on writing modes resulted in stories that were closer to targets. The automatic evaluation shows that the writing modes of text are effectively controlled. To control text generation, we collected **FRIDE** and **FRIDE-XL**, datasets of narrative text annotated with writing modes, which we released to help facilitate further research in writing modes and fine-grained control for storytelling.

Acknowledgments

This paper is based on results obtained from a project JPNP20006, commissioned by the New Energy and Industrial Technology Development Organization (NEDO). This work was partially supported by JSPS KAKENHI Grant Number JP19H05692.

References

[1]Jian Guan, Xiaoxi Mao, and et al. Long text generation by modeling sentence-level and discourse-level coherence. **ACL/IJCNLP**, 2021.

[2]Fuli Luo, Damai Dai, and et al. Learning to control the fine-grained sentiment for story ending generation. In **ACL**, 2019.

[3]Xiangzhe Kong, Jialiang Huang, Ziquan Tung, Jian Guan, and Minlie Huang. Stylized story generation with style-guided planning. **ACL/IJCNLP**, 2021.

[4]Danyang Liu, Juntao Li, Meng-Hsuan Yu, Ziming Huang, Gongshen Liu, Dongyan Zhao, and Rui Yan. A character-centric neural model for automated story generation. In **AAAI**, 2020.

[5]M. Klaassen. **Fiction-Writing Modes: Eleven Essential Tools for Bringing Your Story to Life**. Bookbaby, 2015.

[6]Xiaofei Sun, Chun Fan, Zijun Sun, Yuxian Meng, Fei Wu, and Jiwei Li. Summarize, outline, and elaborate: Long-text generation via hierarchical supervision from extractive summaries. **CoRR**, 2020.

[7]Jessica Morrell. **Between the lines: Master the subtle elements of fiction writing**. Penguin, 2006.

[8]Stephen Halliwell and Aristotle. **Aristotle's Poetics**. University of Chicago Press, Chicago, 1998.

[9]J Devlin, MW Chang, K Lee, and KB Toutanova. Pre-training of deep bidirectional transformers for language understanding. **ACL**, 2019.

[10]Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. Xlnet: Generalized autoregressive pretraining for language understanding. **NeurIPs**, 2019.

[11]Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized BERT pretraining approach. **CoRR**, 2019.

[12]Mike Lewis, Yinhan Liu, and et al. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. **ACL**, 2019.

[13]Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. Exploring the limits of transfer learning with a unified text-to-text transformer. **J. Mach. Learn. Res.**, 2020.

[14]Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. 2018.

[15]Wei Wang, Piji Li, and Hai-Tao Zheng. Consistency and coherency enhanced story generation. In **ECIR**, 2021.

[16]Elizabeth Clark and Noah A. Smith. Choose your own adventure: Paired suggestions in collaborative writing for evaluating story generation models. In **NAACL**, 2021.

[17]Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. STORIUM: A dataset and evaluation platform for machine-in-the-loop story generation. In **EMNLP**, 2020.

[18]Jordan Clive, Kris Cao, and Marek Rei. Control prefixes for text generation. **CoRR**, 2021.

[19]Wojciech Kryscinski, Nazneen Fatema Rajani, and et al. Booksum: A collection of datasets for long-form narrative summarization. **CoRR**, 2021.

[20]Sumanth Dathathri, Andrea Madotto, and et al. Plug and play language models: A simple approach to controlled text generation. In **ICLR**, 2020.

[21]Le Fang, Tao Zeng, Chaochun Liu, Liefeng Bo, Wen Dong, and Changyou Chen. Outline to story: Fine-grained controllable story generation from cascaded events. **CoRR**, 2021.

[22]Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In **ACL**, 2002.

[23]Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In **Text Summarization Branches Out**, 2004.

[24]Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. Bertscore: Evaluating text generation with BERT. In **ICLR**, 2020.