Plug-and-Play Controller for Story Completion: A Pilot Study toward Emotion-aware Story Writing Assistance

Yusuke Mori[1] and **Hiroaki Yamane**[2,1] and **Ryohei Shimizu**[1] and **Tatsuya Harada**[1,2]
[1]The University of Tokyo
[2]RIKEN
{mori, yamane, shimizu, harada}@mi.t.u-tokyo.ac.jp

Abstract

Emotions are essential for storytelling and narrative generation, and as such, the relationship between stories and emotions has been extensively studied. The authors of this paper, including a professional novelist, have examined the use of natural language processing to address the problems of novelists from the perspective of practical creative writing. In particular, the story completion task, which requires understanding the existing unfinished context, was studied from the perspective of creative support for human writers, to generate appropriate content to complete the unfinished parts. It was found that unsupervised pre-trained large neural models of the sequence-to-sequence type are useful for this task. Furthermore, based on the plug-and-play module for controllable text generation using GPT-2, an additional module was implemented to consider emotions. Although this is a preliminary study, and the results leave room for improvement before incorporating the model into a practical system, this effort is an important step in complementing the emotional trajectory of the story.

1 Introduction

In this study, the authors, one of whom is a professional novelist, examined the use of natural language processing to solve the problems faced by novelists from the perspective of practical creative writing. Among the diverse topics related to automatic storytelling and human creativity, **"emotion"** should be emphasized as an important keyword. The relationship between stories and emotions has been an essential part of the research in the field of humanities, especially in the cognitive and affective science of literature (Hogan, 2006; Pandit and Hogan, 2006; Johnson-Laird and Oatley, 2008; Hogan, 2010, 2019).

In providing practical knowledge for authors, creative techniques emphasize the importance of being conscious of readers' emotions (Field, 2006; Snyder, 2005). The theory of the **emotional arc**, which states that a good story can be typified by emotional movement, is well known from the introduction by a popular American novelist, Vonnegut (1995). As presented in Reagan et al. (2016), studies have been conducted to reveal the close relationship between emotions and stories.

Ackerman and Puglisi (2012) insisted that a key component of every character is emotion. In the context of serious storytelling, Lugmayr et al. (2017) insisted that a fundamental aspect of storytelling is emotions, that is, the cognitive aspects that the story evokes in its audience. Numerous efforts have been made to disclose the mystery of the relationship between emotions and stories (Anderson and McMaster, 1982; Strapparava and Mihalcea, 2008; Abdul-Mageed and Ungar, 2017; Kim and Klinger, 2018, 2019a,b; Zad and Finlayson, 2020).

This study focuses on introducing emotions into a story completion (SC) task. The basic task setting in SC is shown in Figure 1.[1] In the field of story generation and understanding, Wang and Wan (2019) proposed SC. We believe that the artificial intelligence (AI) ability to solve SC tasks is important in the context of providing creative support. If writers cannot complete a story and do not know how to proceed with a plot, a suitable model can provide them with appropriate support.

The main contributions of this study are as follows:

- The importance of emotion in stories was confirmed from the perspective of a professional writer, based on which, the possibility of incorporating emotions into SC tasks is discussed for creative support, and a specific method is proposed to accomplish this.

[1]The original story in this figure is from ROCStories (storyid: 0bb3f8b6-117c-45d0-861f-d9953ccc7ddb; storytitle: Dancing).

Proceedings of the First Workshop on Intelligent and Interactive Writing Assistants (In2Writing 2022), pages 46 - 57

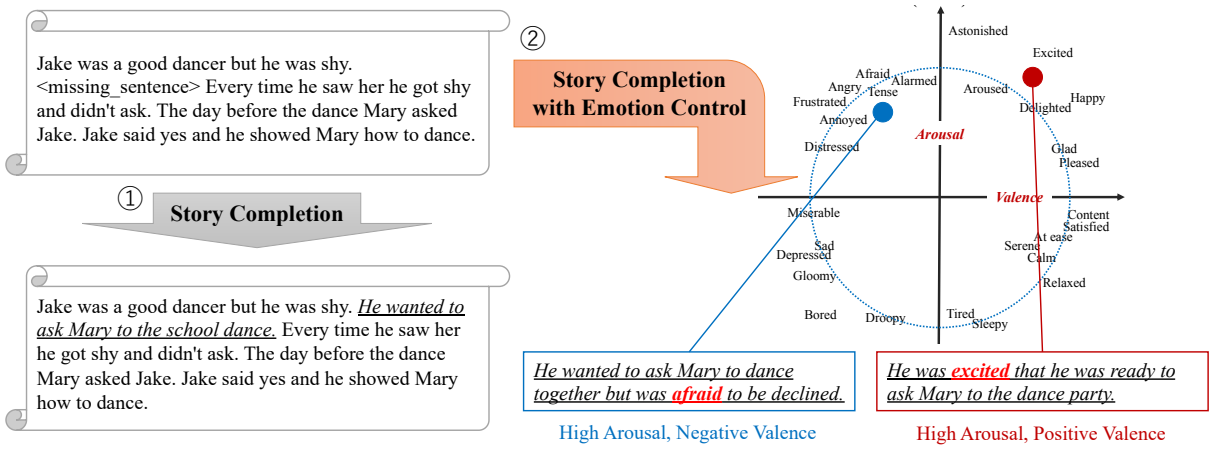

Figure 1: Conceptual diagram of the functionality this study aims for. ① Overview of the story completion task. To address the `<missing_position>` token in an incomplete story, unsupervised pre-trained large neural models are used. ② PPLM is used to control the emotions of the generative text. The representation of the emotions in this figure was reconstructed from an image by Russell (1980).

- Control of SC was examined through our implementation using the plug-and-play language model (PPLM) (Dathathri et al., 2020), whereby the application of the PPLM, which is originally limited, was expanded.

This study is a preliminary study, and the results should be improved before incorporating the model into a practical system. However, we believe that this effort is an important step toward complementing the emotional trajectory of the story and worth discussing for future directions.

As a complementary contribution to this study, we would like to note that a professional writer researched how to use natural language processing (NLP) technology to reflect the viewpoints of writers and researchers. We expect that this work will contribute to building a bridge toward collaborative work between professional writers and researchers in NLP and human computer interface (HCI) to accelerate research in the field of story writing assistance.

2 Related Work

2.1 Story Completion

In the field of story generation and understanding, Wang and Wan (2019) proposed SC. Given any four sentences in a five-sentence story, the objective of the task is to generate a sentence that is not provided (missing plot), to complete the story. In addition to this, research on text infilling has been actively conducted in recent years (Ippolito et al., 2019; Donahue et al., 2020; Huang et al., 2020; Wang et al., 2020). We pointed out that the ability to solve an SC task is essential from the viewpoint of creative support for writers (Mori et al., 2020). If writers cannot complete a story and do not know how to proceed with the plot, AI can provide appropriate support for filling in the blanks.

In this study, controlled text generation with emotion awareness is applied to SC. Focusing on stories, a method is proposed to handle this task in a simple manner by including a special token, specific to the task. By organizing the task in a simple manner, it becomes possible to solve it in a similar way with various models.

2.2 Emotion-aware Storytelling

Some studies have attempted to control story generation by considering emotions (Chandu et al., 2019; Luo et al., 2019; Brahman and Chaturvedi, 2020; Dathathri et al., 2020; Xu et al., 2020). The study closest to ours is that of Brahman and Chaturvedi (2020). They insisted that their study was the first to model the emotional trajectory of the protagonist in neural storytelling. There are significant differences between their study and ours with respect to task setting and the approach taken.

First, Brahman and Chaturvedi (2020) attempted to generate an entire story from the task, while our focus is on the SC task that a model reads to understand what is written in the original context. In this study, dimensional emotions (valence and arousal) were used instead of categorical emotions (four basic emotions in addition to neutral). Dividing emotions into categories is easy to understand,

but for precise control, it is desirable to handle emotions as continuous values. Luo et al. (2019) tackled fine-grained emotion control of story generation, but their objective was story ending rather than completion. Moreover, the controlled emotion was restricted to one dimension (positive-negative). The interest in this study is the control of more diverse two-dimensional emotions based on Russell's circumplex model (Russell, 1980).

2.3 Controllable text generation with Transformer

There are some works in unsupervised pre-trained large neural models for control text generation. Keskar et al. (2019) proposed CTRL to control specific aspects of text generation in large-scale language models. Based on the large-scale language model MEGATRON (Shoeybi et al., 2020) and knowledge-enhanced story generation (Guan et al., 2020), Xu et al. (2020) proposed MEGATRON-CNTRL. In other studies, Rashkin et al. (2020) proposed the task of outline-conditioned story generation, whereby the input only provided a rough sketch of the plot. Therefore, models must generate a story by interweaving the key points provided in the outline. Inspired by plug-and-play generative networks (PPGN) (Nguyen et al., 2017) in computer vision, Dathathri et al. (2020) proposed PPLM, an alternative approach for controlled text generation. Their approach uses attachment models for pre-trained GPT-2 (Radford et al., 2019) to control the word probability distribution during the word-by-word generation process. Optimization is performed ex post facto in the activation space; therefore, no retraining or fine-tuning of the core language model is required. Following this approach, methods have been presented to control the output by adding modules for output control without modifying the core model, such as DELOREAN (DEcoding for nonmonotonic LOgical REAsoNing) (Qin et al., 2020), side-tuning (Zhang et al., 2020a), auxiliary tuning (Zeldes et al., 2020), and GeDi (Krause et al., 2021).

In this study, PPLM, which is a well-designed, simple, and powerful method, is applied for emotion-controllable story generation. Dathathri et al. (2020) explored controlled generation for assistive story writing, demonstrating the usefulness of PPLM in this area. However, they conducted an exploration of open-ended story generation, not SC.

3 Methods

This section describes the proposed method in detail, emphasizing the ingenuity of its implementation. The proposed model has a novel architecture composed of two main parts for SC tasks.

- Fine-tuning unsupervised pre-trained large neural models for the SC task.
- Emotion-aware controlling of fine-tuned models using PPLM.

Studies on applying unsupervised pre-trained large neural models for text infilling have been actively conducted recently (Ippolito et al., 2019; Donahue et al., 2020; Huang et al., 2020; Wang et al., 2020). The first part of our method follows this trend and is verified using various models.

In Subsection 3.2, a modified version of PPLM (Dathathri et al., 2020) is proposed for emotion-aware SC. PPLM, given a prompt (user input text), generates subsequent sentences, as it uses GPT-2 as a base model and tiny attribute models. In this study, the PPLM model was expanded through concatenation with other models.

The model code was implemented using PyTorch (Paszke et al., 2019), which is an open-source machine-learning framework provided as a Python library.[2] To make use of unsupervised pre-trained large neural models, our code was also based on Huggingface Transformers (Wolf et al., 2020), which provide general-purpose architectures for natural language understanding (NLU) and natural language generation (NLG).

The focus here is mainly on Seq2Seq language models (Seq2SeqLMs). For Seq2SeqLMs and its variants, the models below were used.

- BART (Lewis et al., 2020) - BART base, BART large
- T5 (Raffel et al., 2020) - T5 base, T5 large
- PEGASUS (Zhang et al., 2020b) - PEGASUS large
- ProphetNet (Qi et al., 2020) - XLM-ProphetNet large [3]

[2] https://pytorch.org/

[3] We used XLM-ProphetNet because only "uncased" models of ProphetNet were available for pre-trained models. Hence, XLM-ProphetNet, specifically, "microsoft/xprophetnet-large-wiki100-cased," which is a cased version, was used.

Model		#layers	#hidden units	#multi-attention heads
BART (Lewis et al., 2020)	base	6	768	12
	large	12	1024	16
T5 (Raffel et al., 2020)	base	6	768	12
	large	12	1024	16
PEGASUS	large	16	1024	16
ProphetNet	XLM-ProphetNet large	12	1024	16

Table 1: Details of pre-trained models. The Seq2SeqLM in this study consists of encoders and decoders, both having the same number of layers, as indicated in the table for each.

Causal language models (CLMs), which have a left-to-right architecture, do not seem to perform well on SC because they were originally designed for the generation of a continuation of the given prompt and not for completing the missing part, by considering the before and after of the missing part. However, Donahue et al. (2020) proposed the infilling by language modeling (ILM), an approach that enables CLMs to leverage the entire context for text infilling. We left it for future work to apply CLMs to controllable story completion with our proposed method.

PyTorch version 1.11.0, and HuggingFace Transformers version 4.18.0 were used.[4] The details of pre-trained models are displayed in Table 1.

3.1 No-emotion-aware baselines

Initially, models for SC that do not consider emotions should be trained for plug-and-play control. In this study, these methods are referred to as "No-emotion-aware baselines." As shown in Figure 1, a special token was defined for the SC task: "<missing_position>". A special token is inserted into the missing position k, such that the input to the model becomes $S' = \{s_1, ..., s_{k-1},$ <missing_position>$, s_{k+1}, ..., s_n\}$. s stands for a sentence, and the subscript number indicates the position of the sentence in the entire text. Subsequently, the model outputs s_k, as defined in the task.

For Seq2SeqLMs, the S' are concatenated into one text and fed to the encoder. The encoder then passes the calculated embeddings to the decoder and generates text. The output is expected to be a single sentence; however, it was also explored if the model could learn from fine-tuning, including "generate only one sentence," constraints.

[4]We plan to make our code publicly available at https://github.com/mil-tokyo/controllable-story-completion-pilot-study.

3.2 Emotion Controlling Methods

In this study, PPLM was updated for use in emotion control during story completion. PPLM was originally implemented as an additional module for GPT-2 (the default model was GPT2-medium). Adapting PPLM to Seq2SeqLMs required some implementation ingenuities. PPLM was originally designed to generate the continuation of a given text using a decoder-only model. In contrast, in this study, the given text is first processed with the encoder, and then the resulting tensor is used to generate sentences with the decoder.

PPLM has two types of attribute models: bag-of-words (PPLM-BoW) and discriminator (PPLM-Discrim). Originally, PPLM-BoW did not include an emotion control set. PPLM-Discrim has a pre-trained model for sentiment control, but it is positive–negative. In this study, the focus was on PPLM-BoW because it can function by preparing a list of words without additional learning. Thus, the original word list provided in PPLM can be used, but this does not consider valence and arousal. Hence, the NRC valence, arousal, and dominance lexicon (Mohammad, 2018) (NRC-VAD lexicon) was used to obtain the word list annotated with dimensional emotion values, which was subsequently fed into PPLM-BoW. Instead of using the entire NRC-VAD lexicon as is, in our implementation, a range of values can be specified for valence and arousal (and dominance) at runtime to obtain a subset within that range.

4 Experimental Setup

4.1 Dataset

In this pilot study, the proposed method was trained and evaluated using ROCStories (Mostafazadeh et al., 2016). As shown in Table 2, the dataset was randomly split in a ratio of 8:1:1 to obtain training, development, and test sets. One sentence was removed from the five-sentence story. The missing position k was randomly determined based on a

set	#stories	how to give k
Training	78,528	randomly during training
dev	9,816	when creating a dataset
Test	9,817	when creating a dataset
total	98,161	

Table 2: Overview of the dataset used.

discrete uniform distribution. For the development and test sets, the removal procedure was performed when creating the dataset to improve reproducibility. For the training set, the original five-sentence story was retained in the dataset and a sentence was randomly removed while reading the data during training. This setting followed that of our previous study (Mori et al., 2020).

4.2 Training Details

For training, the AdamW (Loshchilov and Hutter, 2019) optimizer was used with parameters $\beta_1 = 0.9, \beta_2 = 0.999, and \epsilon = 1e-08$. The initial learning rate was set to $3e-05$ and linearly decreased thereafter from the initial point to 0 to avoid overfitting. The model was fine-tuned using NVIDIA Tesla V100 GPUs and the size of the training batch was set to 8.

We use two sets of training parameters. One is task-specific parameters, defined for each model based on with reference to its use for the summarization task. The other is common parameters for all models.

Seq2SeqLMs significantly improved the performance compared to conventional models in text-to-text tasks, especially in summarization and translation. Of these two well-worked tasks, we hypothesized that the training settings for summarization are closer to what we need for SC. SC requires methods to understand the context, to generate appropriate sentences for completion. The given context is typically longer than a sentence for completion. In summary, methods are required to understand the entire text, to generate shorter sentences to represent it. Although there are two types of approaches, extractive summarization and abstractive summarization, the basic objective is the same. On the other hand, in translation tasks, although it is also important to understand the input content, the output length is not significantly different from the input length (note that there is a difference related to the nature of each language). There are also application examples, such as paraphrasing in one language, but the input and output are generally in different languages during translation.

What varies from model to model is the setting such as length penalty and max length of input and output sequence. The length penalty places a constraint on the length of the generated sentences, prompting the generation of longer sentences if it is greater than 1.0, and shorter sentences if it is less than 1.0. As mentioned above, task-specific parameters prepared for summarization were used in this study. This was done to ensure the fairness of the settings by unifying the parameters in "solving SC by directly applying the settings of the summarization task." [5] For this reason, the length penalty was set to 2.0 for T5 in this experiment, 1.0 for BART, and 0.8 for PEGASUS. For XLM-ProphetNet, the penalty was 2.0.

For a different sense of fairness, we provided another setting that uses a common length penalty. In this setting, the length penalty is 1.0.

4.3 Evaluation Metrics

It is necessary to evaluate a large number of models and their variants (model parameters, training parameters, tasks that are fine-tuned beforehand, etc.). Thus, automatic evaluation metrics were employed instead of human evaluation. Stories entertain the reader (or evoke other emotions); therefore, human evaluation is important. However, there is a huge cost involved in terms of time and money for evaluating various parameters in many models. In addition, there are factors such as age, gender, and regional trends in texts, particularly in stories. The problem is that stories liked by someone are not always liked by others. In this section, the focus is on automatic evaluation metrics for a large number of models. The human evaluation of a narrowed-down list of promising candidate models is left for future work.

The following metrics were used for the evaluation: BLEU (Papineni et al., 2002), ROUGE (Lin, 2004), METEOR (Banerjee and Lavie, 2005), BERTScore (Zhang* et al., 2020),[6] and BLEURT (Sellam et al., 2020).[7] The Python library HuggingFace Datasets was used for certain metrics;

[5] There is no generic parameter for the "summarization task" for PEGASUS, so the parameter for summarization of the XSUM dataset was used.

[6] https://github.com/Tiiiger/bert_score

[7] https://github.com/google-research/bleurt

'sacrebleu' as BLEU, ROUGE and METEOR.[8] For each of BERTScore and BLEURT, the original implementation of each paper was used.

5 Results

5.1 No-emotion-aware baselines

First, experiments were conducted using no-emotion-aware baselines. Table 3 lists the test set results of Seq2SeqLMs evaluated using automatic evaluation metrics. In this comparison, the entire story was not compared; however, the generated complementary sentence was compared with the original sentence (the missing sentence). The value of F1 was used for ROUGE and BERTScore. In addition, for BERTScore, the authors obtained an average when evaluating the models.[9] BLEURT was treated in a similar manner.

The results indicated that BART large exhibited the highest scores for every metric. For a deeper analysis of the metric results, Table 4 was created for average generation length and runtime. In BART base, BART large, and PEGASUS, the two training settings didn't have a significant impact. On the other hand, for T5 base, T5 large, and XLM-ProphetNet, better results were obtained when using task-specific parameters. The result suggests that the parameters for summarization work well for story completion, especially when the model requires a large length penalty for summarization tasks.

Table 5 and 6 display the examples generated.

5.2 Emotion Controlling Method

The Seq2SeqLM + PPLM-BoW results are presented in Table 7. As BART large displayed the best result in the no-emotion-aware baseline experiment, BART large was used as the first step of Emotion-aware SC with Seq2SeqLM + PPLM.

In the examples shown in Table 7, the ranges of valence and arousal were set to $0.0 <= valence <= 0.3$ and $0.7 <= arousal <= 1.0$, respectively. As valence is negative and arousal is high, negative and excited emotions are expected to emerge. The results of an uncontrolled trial (unperturbed) and three controlled trials (perturbed) are presented as examples. Perturbed 1 seems to be controlled by "negative and excited." In the context of careful driving, it is not unnatural for events related to the car to occur, and on top of that, the expression that the car gets stuck is negative. We showed an example where the generation of emotion-controlled sentences worked well. However, the adjustment of the parameters to generate a sequence was very severe. PPLM provides parameters to manipulate the generated results, but it is very difficult to adjust these parameters, at least in combination with Seq2SeqLM.

We should note that the BART large model used here was trained with an older version of PyTorch and Transformers. Unfortunately, the version trained with PyTorch 1.11.0 and Transformers 4.18.0 used in this Seq2SeqLM Story Completion did not produce good results with the same generation parameters. Although we could run the modified PPLM with the libraries of the newer version, the choice of the fine-tuned model is also severe.

PPLM was originally designed for use with GPT-2, but in this study, it was modified and applied to Seq2SeqLM. Specifically, it was confirmed that PPLM works on BART. However, when we used the Seq2SeqLM model which was fine-tuned for no-emotion-aware SC to generate sentences controlled with PPLM, we found that the sentences tended to be shorter than those generated without PPLM.

6 Discussion

The no-emotion-aware baseline results indicate that BART large exhibited the highest scores for every metric. In this study, we used two sets of training parameters: one is based on summarization task-specific parameters and the other is common parameters. The result showed that the parameters for summarization work well for story completion, compared to common parameters that do not account for differences between models. Future studies should search for specific parameters for each model that are more suitable for SC.

In this study, PPLM was extended and combined with BART, a representative model of Seq2SeqLMs. In addition, by combining PPLM with the NRC-VAD lexicon, a basis was created for SC to consider valence and arousal. However, there is still a lot of room for improvement in the results.

In text generation, it is important to control the behavior of the model using parameters such as

[8] https://github.com/huggingface/datasets

[9] https://github.com/Tiiiger/bert_score/blob/master/example/Demo.ipynb

	BLEU	ROUGE-1	ROUGE-2	ROUGE-L	METEOR	BERTScore	BLEURT
BART base w/ specific param	5.352848	0.265496	0.082603	0.245470	0.254414	0.909720	-0.432042
BART large w/ specific param	7.390772	0.291679	0.106530	0.271545	0.279876	0.914704	-0.373194
PEGASUS large w/ specific param	5.401445	0.265151	0.085482	0.243784	0.266451	0.909168	-0.443984
T5 base w/ specific param	4.390108	0.253425	0.070985	0.232174	0.244871	0.907397	-0.473313
T5 large w/ specific param	6.249401	0.282742	0.095236	0.259644	0.276074	0.912142	-0.404434
XLM-ProphetNet large w/ specific param	0.116252	0.159532	0.010753	0.148529	0.065040	0.853637	-0.821382
BART base	5.352848	0.265651	0.082704	0.245416	0.254414	0.909720	-0.432042
BART large	7.390772	0.291414	0.106375	0.271576	0.279876	0.914704	-0.373194
20220410_003_pegasus_large	5.401445	0.265209	0.085513	0.243719	0.266451	0.909168	-0.443984
T5 base	2.330794	0.257133	0.074025	0.241255	0.194306	0.900627	-0.911796
T5 large	2.332709	0.288103	0.098576	0.270357	0.225574	0.903646	-0.912072
XLM-ProphetNet large	0.071638	0.158260	0.009964	0.146465	0.064679	0.852067	-0.798809

Table 3: The result of no-emotion-aware Seq2SeqLMs evaluated with automatic evaluation metrics.

	BLEU	generated length	runtime	samples/sec
BART base w/ specific param	5.3528	14.5	344.5440	-0.003
BART large w/ specific param	7.3907	15.0	546.4531	-0.002
PEGASUS large w/ specific param	5.4014	13.6	890.2809	-0.001
T5 base w/ specific param	4.3901	14.9	595.7259	-0.002
T5 large w/ specific param	6.2494	14.7	1031.0659	-0.001
XLM-ProphetNet large w/ specific param	0.1163	10.8	960.6619	-0.001
BART base	5.3528	14.5	352.5765	-0.003
BART large	7.3907	15.0	556.1080	-0.002
20220410_003_pegasus_large	5.4014	13.6	893.2609	-0.001
T5 base	2.3308	13.8	487.8538	-0.002
T5 large	2.3327	13.6	866.5806	-0.001
XLM-ProphetNet large[10]	0.0716	9.0	11589.1036	-0.000

Table 4: The mean generated length and the runtime of no-emotion-aware Seq2SeqLMs. "w/ specific param" indicates that the model is trained using the task-specific parameters of each model.

storyid	dc36af5e-a65f-4193-8f3c-5162c8af6755
context	<missing_position> I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
missing_id	0
GT	I went to a restaurant yesterday.
BART base	I went to the fish market with my friends.
BART large	I went to the fish market yesterday.
PEGASUS large	I went to the fish market today for the first time.
T5 base	I went to a fish market one day. I was very hungry.
T5 large	I went to a fish market one day with my friends.
XLM-ProphetNet large	She was to to the....
GT completed story	I went to a restaurant yesterday. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
BART base completed story	I went to the fish market with my friends. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
BART large completed story	I went to the fish market yesterday. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
PEGASUS large completed story	I went to the fish market today for the first time. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
T5 base completed story	I went to a fish market one day. I was very hungry. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
T5 large completed story	I went to a fish market one day with my friends. I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.
XLM-ProphetNet large completed story	She was to to the.... I wanted to take out some fish. But then the lady was not using gloves. I was disgusted. I ended up walking out.

Table 5: Examples of contexts and completion sentences generated by no-emotion-aware Seq2SeqLMs. In this case, the task-specific parameters for each model were used.

the length penalty. Two types of parameters were experimented with in this study, but further effort is required to determine the best parameter. The optimal hyperparameters seem to be naturally different for each model. It is not realistic to check all outputs using the human eye while adjusting hyperparameters within a wide range of values for many models. Therefore, an automatic evaluation

storyid	f2a013bd-852f-43f4-9012-4db8ae44c64e
context	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. <missing_position> Jane didn't care as she knew she was making him feel better.
missing_id	3
GT	This would look strange to the public.
BART base	One day, her dog fell down and broke his leg.
BART large	Her dog got very sick and couldn't run anymore.
PEGASUS large	One day, her dog got sick and had to be put down.
T5 base	One day, she noticed that her dog was very sick.
T5 large	One day, her dog got sick and couldn't walk.
XLM-ProphetNet large	He was to to the the..
GT completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. This would look strange to the public. Jane didn't care as she knew she was making him feel better.
BART base completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. One day, her dog fell down and broke his leg. Jane didn't care as she knew she was making him feel better.
BART large completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. Her dog got very sick and couldn't run anymore. Jane didn't care as she knew she was making him feel better.
PEGASUS large completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. One day, her dog got sick and had to be put down. Jane didn't care as she knew she was making him feel better.
T5 base completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. One day, she noticed that her dog was very sick. Jane didn't care as she knew she was making him feel better.
T5 large completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. One day, her dog got sick and couldn't walk. Jane didn't care as she knew she was making him feel better.
XLM-ProphetNet large completed story	Jane had a very sick dog. Her dog was old and couldn't run anymore. So that he still felt young, Jane used to walk her dog in a pram. He was to to the the.. Jane didn't care as she knew she was making him feel better.

Table 6: Examples of contexts and completion sentences generated by no-emotion-aware Seq2SeqLMs. In this case, the same hyperparameters were used for length penalty and max length.

Context	I got a call from the hospital. My doctor told me to stop everything I'm doing and come to her. Although I was nervous, I tried to drive calmly. <missing_sentence> The doctor diagnosed me with leukemia.
missing sentence	The front desk worker sent me to an office.
Unperturbed	However, my blood.ItItMy
Perturbed 0	However, the car..
Perturbed 1	My car got stuck...
Perturbed 2	.....

Table 7: An example of emotion-controlled SC with BART large + PPLM-BoW (0.0 <= Valence <= 0.3 and 0.7 <= Arousal <= 1.0).

mechanism is required.

The application of these methods to other datasets is left for future work. As a representative example, the WritingPrompts dataset (Fan et al., 2018) was considered. Stories in WritingPrompts vary in terms of length; therefore, the importance of a single sentence varies from one story to the other. With very long stories, generally trimming is used to retain a predetermined number of words from the start while truncating the rest. Hence, this dataset was not considered to be suitable for the SC tasks for now. Thus, as a starting point, ROCStories was adopted.

7 Considerations by a Professional Writer

As noted in the Introduction, one of the authors of this study was a professional novelist. This work is a collaborative effort between researchers and a professional creative writer. More precisely, the first author of this paper is a professional Japanese novelist as well as a researcher in the field of story understanding and generation.

In Section 6, the viewpoint of the researchers is discussed. In this section, the positioning and prospects of this study are discussed from the novelist's perspective.

In an experiment conducted separately from this study, four professional creative writers were asked to evaluate a creative writing support system.[11] The results of that experiment confirmed that there might be a negative perception of the system's ability to control the output if there are parameters with which the user is not familiar. Although it would be desirable for users to have the freedom to adjust the outcome, too many parameters make them lost. They do not know what to do, resulting in confusion on the user's part in using the system and in a negative impression.

As previously mentioned, our modified PPLM for controllable SC addressed in this study is difficult to adjust. Moreover, in its current state, users are required to understand what "valence" and "arousal" mean. We believe that treating both dimensions rather than one dimension (positive-negative) would be important for future directions in this area, but this idea is not yet widespread. Hence, it is difficult for this approach to provide professional writers with the desired results for now. At this point, there was concern that other professional writers would have a negative impression of the "creative writing support system that controls the emotions of the generated text" as a whole. That is why no human evaluation was conducted on this study, except by the novelist author.

For practitioners, the extent to which AI could replace their own work is an important issue; there is also concern that it could trigger a sense of avoidance toward AI. Prudence is needed in conducting research, and professional evaluations, which are important topics of discussion.

Some professional novelists write from beginning to end in order, while others come up with certain parts but cannot come up with the correct sentences to fill in the gaps. SC is an important task in helping the latter. From the creative writer's perspective, it is helpful to have a system that understands the meaning of one's own writing and then fills in the missing parts. Furthermore, as the importance of the emotional arc in a story becomes increasingly apparent, a system that controls the output of the emotions desired by the user as well as an evaluation index that considers emotions would be helpful.

[11]The details of the human evaluation consist the part of the doctoral dissertation of the first author. The dissertation will be publicly available in the UTokyo Repository, https://repository.dl.itc.u-tokyo.ac.jp/.

8 Conclusion

In this study, the SC task was considered for various emotions. Previous studies on emotion-aware story generation have restricted emotions to one dimension (positive-negative) or categorical ones. Our aim was to control more diverse emotions, so the issue of two-dimensional control was addressed based on Russell's circumplex model.

Our implementation made it possible to control SC using PPLM. This expands the application of PPLM, which was originally limited to the task of "generating the continuation of a prompt." Although the goal of controlling emotions was accomplished, it was difficult to adjust the parameters. Whether this difficulty in coordination can be improved through innovative implementation or demands a completely different approach requires further examination.

Acknowledgements

We would like to thank Yusuke Mukuta for the helpful discussions. This work was partially supported by JST AIP Acceleration Research JPMJCR20U3, Moonshot R&D Grant Number JPMJPS2011, JSPS KAKENHI Grant Number JP19H01115, and JP20H05556 and Basic Research Grant (Super AI) of Institute for AI and Beyond of the University of Tokyo.

References

Muhammad Abdul-Mageed and Lyle Ungar. 2017. Emonet: Fine-grained emotion detection with gated recurrent neural networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 718–728, Vancouver, Canada. Association for Computational Linguistics.

Angela Ackerman and Becca Puglisi. 2012. *The Emotion Thesaurus: A Writer's Guide to Character Expression*. JADD Publishing.

C. W. Anderson and G. E. McMaster. 1982. Computer assisted modeling of affective tone in written documents. *Computers and the Humanities*, 16(1):1–9.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Ex-*

trinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Khyathi Chandu, Shrimai Prabhumoye, Ruslan Salakhutdinov, and Alan W Black. 2019. "my way of telling a story": Persona based grounded story generation. In *Proceedings of the Second Workshop on Storytelling*, pages 11–21, Florence, Italy. Association for Computational Linguistics.

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*.

Chris Donahue, Mina Lee, and Percy Liang. 2020. Enabling language models to fill in the blanks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2492–2501, Online. Association for Computational Linguistics.

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, Melbourne, Australia. Association for Computational Linguistics.

Syd Field. 2006. *The Screenwriter's Workbook, Revised Edition*. Delta Trade Paperbacks.

Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pretraining model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.

Patrick Colm Hogan. 2006. Narrative universals, heroic tragi-comedy, and shakespeare's political ambivalence. *College Literature*, 33(1):34–66.

Patrick Colm Hogan. 2010. A passion for plot: Prolegomena to affective narratology. *symplokē*, 18(1-2):65–81.

Patrick Colm Hogan. 2019. Description, explanation, and the meanings of "narrative". *Evolutionary Studies in Imaginative Culture*, 3:45+. 1, 45, Critical essay.

Yichen Huang, Yizhe Zhang, Oussama Elachqar, and Yu Cheng. 2020. INSET: Sentence infilling with INter-SEntential transformer. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2502–2515, Online. Association for Computational Linguistics.

Daphne Ippolito, David Grangier, Chris Callison-Burch, and Douglas Eck. 2019. Unsupervised hierarchical story infilling. In *Proceedings of the First Workshop on Narrative Understanding*, pages 37–43, Minneapolis, Minnesota. Association for Computational Linguistics.

R. N. Johnson-Laird and Keith Oatley. 2008. *Emotions, music, and literature.*, Handbook of emotions, 3rd ed., pages 102–113. The Guilford Press, New York, NY, US.

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*.

Evgeny Kim and Roman Klinger. 2018. Who feels what and why? annotation of a literature corpus with semantic roles of emotions. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1345–1359, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Evgeny Kim and Roman Klinger. 2019a. An analysis of emotion communication channels in fan-fiction: Towards emotional storytelling. In *Proceedings of the Second Workshop on Storytelling*, pages 56–64, Florence, Italy. Association for Computational Linguistics.

Evgeny Kim and Roman Klinger. 2019b. Frowning Frodo, wincing Leia, and a seriously great friendship: Learning to classify emotional relationships of fictional characters. 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 647–653, Minneapolis, Minnesota. Association for Computational Linguistics.

Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, richard socher, and Nazneen Rajani. 2021. Gedi: Generative discriminator guided sequence generation.

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, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Artur Lugmayr, Erkki Sutinen, Jarkko Suhonen, Carolina Islas Sedano, Helmut Hlavacs, and Calkin Suero Montero. 2017. Serious storytelling - a first definition and review. *Multimedia Tools and Applications*, 76:15707–15733.

Fuli Luo, Damai Dai, Pengcheng Yang, Tianyu Liu, Baobao Chang, Zhifang Sui, and Xu Sun. 2019. Learning to control the fine-grained sentiment for story ending generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6020–6026, Florence, Italy. Association for Computational Linguistics.

Saif Mohammad. 2018. Obtaining reliable human ratings of valence, arousal, and dominance for 20,000 English words. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 174–184, Melbourne, Australia. Association for Computational Linguistics.

Yusuke Mori, Hiroaki Yamane, Yusuke Mukuta, and Tatsuya Harada. 2020. Finding and generating a missing part for story completion. In *Proceedings of the The 4th Joint SIGHUM Workshop on Computational Linguistics for Cultural Heritage, Social Sciences, Humanities and Literature*, pages 156–166, Online. International Committee on Computational Linguistics.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Anh Nguyen, Jeff Clune, Yoshua Bengio, Alexey Dosovitskiy, and Jason Yosinski. 2017. Plug & play generative networks: Conditional iterative generation of images in latent space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. IEEE.

Lalita Pandit and Patrick Colm Hogan. 2006. Introduction: morsels and modules: on embodying cognition in shakespeare's plays (1). *College Literature*, 33:1+. 1, Article.

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 on Association for Computational Linguistics*, pages 311–318, Stroudsburg, PA, USA. Association for Computational Linguistics.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. 2019. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Weizhen Qi, Yu Yan, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. Prophetnet: Predicting future n-gram for sequence-to-sequence pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Lianhui Qin, Vered Shwartz, Peter West, Chandra Bhagavatula, Jena D. Hwang, Ronan Le Bras, Antoine Bosselut, and Yejin Choi. 2020. Back to the future: Unsupervised backprop-based decoding for counterfactual and abductive commonsense reasoning. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 794–805, Online. Association for Computational Linguistics.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. Plotmachines: Outline-conditioned generation with dynamic plot state tracking.

Andrew J. Reagan, Lewis Mitchell, Dilan Kiley, Christopher M. Danforth, and Peter Sheridan Dodds. 2016. The emotional arcs of stories are dominated by six basic shapes. *EPJ Data Science*, 5(1):31.

James A. Russell. 1980. A circumplex model of affect. *Journal of personality and social psychology*, 39:1161–1178.

Thibault Sellam, Dipanjan Das, and Ankur 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, Online. Association for Computational Linguistics.

Mohammad Shoeybi, Mostofa Patwary, Raul Puri, Patrick LeGresley, Jared Casper, and Bryan Catanzaro. 2020. Megatron-lm: Training multi-billion parameter language models using model parallelism.

Blake Snyder. 2005. *SAVE THE CAT! The Last Book on Screenwriting You'll Ever Need*. Michael Wiese Productions.

Carlo Strapparava and Rada Mihalcea. 2008. Learning to identify emotions in text. In *Proceedings of the 2008 ACM Symposium on Applied Computing*, SAC '08, pages 1556–1560, New York, NY, USA. ACM.

Kurt Vonnegut. 1995. Kurt vonnegut on the shapes of stories. https://www.youtube.com/watch?v=oP3c1h8v2ZQ. Video. Accessed: October 17, 2020.

Su Wang, Greg Durrett, and Katrin Erk. 2020. Narrative interpolation for generating and understanding stories.

Tianming Wang and Xiaojun Wan. 2019. T-CVAE: Transformer-based conditioned variational autoencoder for story completion. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence*, pages 5233–5239. International Joint Conferences on Artificial Intelligence Organization.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Peng Xu, Mostofa Patwary, Mohammad Shoeybi, Raul Puri, Pascale Fung, Anima 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*, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Samira Zad and Mark Finlayson. 2020. Systematic evaluation of a framework for unsupervised emotion recognition for narrative text. In *Proceedings of the First Joint Workshop on Narrative Understanding, Storylines, and Events*, pages 26–37, Online. Association for Computational Linguistics.

Yoel Zeldes, Dan Padnos, Or Sharir, and Barak Peleg. 2020. Technical report: Auxiliary tuning and its application to conditional text generation.

Jeffrey O. Zhang, Alexander Sax, Amir Zamir, Leonidas Guibas, and Jitendra Malik. 2020a. Side-tuning: A baseline for network adaptation via additive side networks. In *Computer Vision – ECCV 2020*, pages 698–714, Cham. Springer International Publishing.

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

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