

Persona-Guided Planning for Controlling the Protagonist’s Persona in Story Generation

Zhexin Zhang*, Jiaxin Wen*, Jian Guan, Minlie Huang†

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.
{zx-zhang18, wenjx17, j-guan19}@mails.tsinghua.edu.cn
aihuang@tsinghua.edu.cn

Abstract

Endowing the protagonist with a specific personality is essential for writing an engaging story. In this paper, we aim to control the protagonist’s persona in story generation, i.e., generating a story from a leading context and a persona description, where the protagonist should exhibit the specified personality through a coherent event sequence. Considering that personas are usually embodied implicitly and sparsely in stories, we propose a planning-based generation model named CONPER to explicitly model the relationship between personas and events. CONPER first plans events of the protagonist’s behavior which are motivated by the specified persona through predicting one target sentence, then plans the plot as a sequence of keywords with the guidance of the predicted persona-related events and common-sense knowledge, and finally generates the whole story. Both automatic and manual evaluation results demonstrate that CONPER outperforms state-of-the-art baselines for generating more coherent and persona-controllable stories. Our code is available at <https://github.com/thu-coai/ConPer>.

1 Introduction

Stories are important for entertainment. They are made engaging often by portraying animated and believable characters since a story plot unfolds as the characters interact with the object world created in the story (Young, 2000). Cognitive psychologists determined that the ability of an audience to comprehend a story is strongly correlated with the characters’ believability (Graesser et al., 1991). And the believability mostly depends on whether the characters’ reaction to what has happened and their deliberate behavior accord with their personas (e.g., weakness, abilities, occupations) (Madsen and Nielsen, 2009; Riedl and Young, 2010).

*Equal contribution.

†Corresponding author.

Context: . . . Artur Boruc, a Polish national pilot, was going to get the group infiltrated into the area and prepared for the attack. . . . Metal began to tear through the thin wings outside the small windows and pinged heavily off the underside of the plane, a quiet arrival in Poland wasn’t going to be an option anymore.

Persona A: [Boruc] A skilled pilot, trained in operating flight controls on the most common planes.

Generated Story by CONPER Conditioned on Persona A: . . . He had been doing this much before, almost a week ago. He took a long pull of the airlock, checked his controls, and made a quick mental note of the exact sequence of instructions. He knew that he couldn’t be sure if this would be safe for much longer. . .

Persona B: [Boruc] An unskilled pilot, and never trained in operating flight controls.

Generated Story by CONPER Conditioned on Persona B: . . . He cursed as the plane suffered a complete failure and in a way had caused it to come to a stop. . . . He’d never flown before, so he didn’t know how to pilot in this situation and his experience of the controls had not been good either. . .

Table 1: An example for controlling the protagonist’s persona in story generation. The **Context** and **Persona A** are sampled from the STORIUM dataset (Akoury et al., 2020). The protagonist’s name is shown in the square bracket. And we manually write **Persona B** based on **Persona A**. We highlight the sentences which embody the given personas in red.

Furthermore, previous studies have also stressed the importance of personas in stories to maintain the interest of audience and instigate their sense of empathy and relatedness (Cavazza et al., 2009; Chandu et al., 2019). However, despite the broad recognition of its importance, it has not yet been widely explored to endow characters with specified personalities in story generation.

In this paper, we present the first study to impose free-form controllable persona on story generation. Specifically, we require generation models to generate a coherent story, where the protagonist should exhibit the desired personality. We focus on controlling the persona of only the protagonist of a story in this paper and leave the modeling of personas of multiple characters for future work. As exemplified in Table 1, given a context to present the story settings including characters, location, problems (e.g., “Boruc” was suffering

from a plane crash) and a persona description, the model should generate a coherent story to exhibit the persona (e.g., what happened when “Boruc” was “skilled” or “unskilled”). In particular, we require the model to embody the personality of the protagonist implicitly through his actions (e.g., “checked his controls” for the personality “skilled pilot”). Therefore, the modeling of relations between persona and events is the **first challenge** of this problem. Then, we observe that only a small amount of events in a human-written story relate to personas directly and the rest serve for explaining the cause and effect of these events to maintain the coherence of the whole story. Accordingly, the **second challenge** is learning to plan a coherent event sequence (e.g., first “finding the plane shaking”, then “checking controls”, and finally “landing safely”) to embody personas naturally.

In this paper, we propose a generation model named CONPER to deal with *Controlling Persona* of the protagonist in story generation. Due to the persona-sparsity issue that most events in a story do not embody the persona, directly fine-tuning on real-world stories may mislead the model to focus on persona-unrelated events and regard the persona-related events as noise (Zheng et al., 2020). Therefore, before generating the whole story, CONPER first plans persona-related events through predicting one target sentence, which should be motivated by the given personality following the leading context. To this end, we extract persona-related events that have a high semantic similarity with the persona description in the training stage. Then, CONPER plans the plot as a sequence of keywords to complete the cause and effect of the predicted persona-related events with the guidance of commonsense knowledge. Finally, CONPER generates the whole story conditioned on the planned plot. The stories are shown to have better coherence and persona-consistency than state-of-the-art baselines.

We summarize our contributions as follows:

- I.** We propose a new task of controlling the personality of the protagonist in story generation.
- II.** We propose a generation model named CONPER to impose specified persona into story generation by planning persona-related events and a keyword sequence as intermediate representations.
- III.** We empirically show that CONPER can achieve better controllability of persona and generate more coherent stories than strong baselines.

2 Related Work

Story Generation There have been wide explorations for various story generation tasks, such as story ending generation (Guan et al., 2019), story completion (Wang and Wan, 2019) and story generation from short prompts (Fan et al., 2018), titles (Yao et al., 2019) or beginnings (Guan et al., 2020). To improve the coherence of story generation, prior studies usually first predicted intermediate representations as plans and then generated stories conditioned on the plans. The plans could be a series of keywords (Yao et al., 2019), an action sequence (Fan et al., 2019; Goldfarb-Tarrant et al., 2020) or a keyword distribution (Kang and Hovy, 2020). In terms of character modeling in stories, some studies focused on learning characters’ persona as latent variables (Bamman et al., 2013, 2014) or represented characters as learnable embeddings (Ji et al., 2017; Clark et al., 2018; Liu et al., 2020). Chandu et al. (2019) proposed five types of specific personas for visual story generation. Brahman et al. (2021) formulated two new tasks including character description generation and character identification. In contrast, we focus on story generation conditioned on personas in a free form of text to describe one’s strengths, weaknesses, abilities, occupations and goals.

Controllable Generation Controllable text generation aims to generate texts with specified attributes. For example, Keskar et al. (2019) pre-trained a language model conditioned on control codes of different attributes (e.g., domains, links). Dathathri et al. (2020) proposed to combine a pre-trained language model with trainable attribute classifiers to increase the likelihood of the target attributes. Recent studies in dialogue models focused on controlling through sentence functions (Ke et al., 2018), politeness (Niu and Bansal, 2018) and conversation targets (Tang et al., 2019). For storytelling, Brahman et al. (2020) incorporated additional phrases to guide the story generation. Brahman and Chaturvedi (2020) proposed to control the emotional trajectory in a story by regularizing the generation process with reinforcement learning. Rashkin et al. (2020) generated stories from outlines of characters and events by tracking the dynamic plot states with a memory network.

A similar research to ours is Zhang et al. (2018), which introduced the PersonaChat dataset for endowing the chit-chat dialogue agents with a consis-

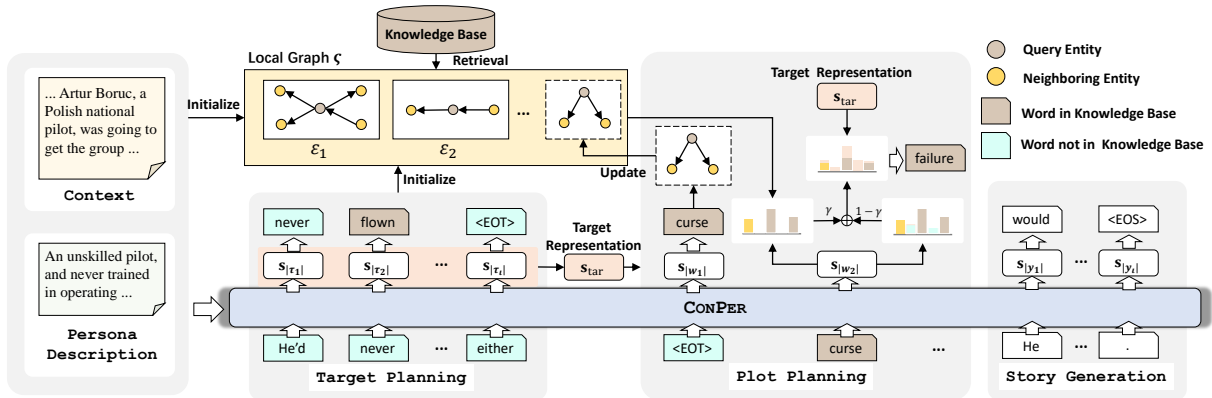


Figure 1: Model overview of CONPER. The training process is divided into the following three stages: (a) Target Planning: planning persona-related events (called “target” for short); (b) Plot Planning: planning a keyword sequence as an intermediate representation of the story with the guidance of the target and a dynamically growing local knowledge graph; And (c) Story Generation: generating the whole story conditioned on the input and plans.

tent persona. However, dialogues in PersonaChat tend to exhibit the given personas explicitly (e.g., the agent says “I am terrified of dogs” for the persona “I am afraid of dogs”). For quantitative analysis, we compute the ROUGE score (Lin, 2004) between the persona description and the dialogue or story. We find that the rouge-2 score is 0.1584 for PersonaChat and 0.018 for our dataset (i.e., STORIUM). The results indicate that exhibiting personas in stories requires a stronger ability to associate the action of a character and his implicit traits compared with exhibiting personas in dialogues.

Commonsense Knowledge Recent studies have demonstrated that incorporating external commonsense knowledge significantly improved the coherence and informativeness for dialog generation (Zhou et al., 2018a; Zhong et al., 2020), story ending generation (Guan et al., 2019), essay generation (Yang et al., 2019), story generation (Guan et al., 2020; Xu et al., 2020; Mao et al., 2019) and story completion (Ammanabrolu et al., 2021). These studies usually retrieved a static local knowledge graph which contains entities mentioned in the input, and their related entities. We propose to incorporate the knowledge dynamically during generation to better model the keyword transition in a long-form story.

3 Methodology

We define our task as follows: given a context $X = (x_1, x_2, \dots, x_{|X|})$ with $|X|$ tokens, and a persona description for the protagonist $P = (p_1, p_2, \dots, p_l)$ of length l , the model should generate a coherent story $Y = (y_1, y_2, \dots, y_{|Y|})$ of

length $|Y|$ to exhibit the persona. To tackle the problem, the popular generation model such as GPT2 commonly employ a left-to-right decoder to minimize the negative log-likelihood \mathcal{L}_{ST} of human-written stories:

$$\mathcal{L}_{ST} = - \sum_{t=1}^{|Y|} \log P(y_t | y_{<t}, S), \quad (1)$$

$$P(y_t | y_{<t}, S) = \text{softmax}(\mathbf{s}_t \mathbf{W} + \mathbf{b}), \quad (2)$$

$$\mathbf{s}_t = \text{Decoder}(y_{<t}, S), \quad (3)$$

where S is the concatenation of X and P , \mathbf{s}_t is the decoder’s hidden state at the t -th position of the story, \mathbf{W} and \mathbf{b} are trainable parameters. Based on this framework, we divide the training process of CONPER into three stages as shown in Figure 1.

3.1 Target Planning

We observe that most sentences in a human-written story do not aim to exhibit any personas, but serve to maintain the coherence of the story. Fine-tuning on these stories directly may mislead the model to regard input personas as noise and focus on modeling the persona-unrelated events which are in the majority. Therefore, we propose to first predict persona-related events (i.e., the target) before generating the whole story.

We use an automatic approach to extract the target from a story since there is no available manual annotation. Specifically, we regard the sentence as the target which has the highest semantic similarity with the persona description. We consider only one sentence as the target in this work due to the persona-sparsity issue, and we also present the

result of experimenting with two sentences as the target in the appendix B.1. More explorations of using multiple target sentences are left as future work. We adopt NLTK (Bird et al., 2009) for sentence tokenization. And we measure the similarity between sentences using BERTScore_{Recall} (Zhang et al., 2019) with RoBERTa_{Large} (Liu et al., 2019) as the backbone model. Let $T = (\tau_1, \tau_2, \dots, \tau_\iota)$ denote the target sentence of length ι , which should be a sub-sequence of Y . Formally, the loss function \mathcal{L}_{TP} for this stage can be derived as follows:

$$\mathcal{L}_{TP} = - \sum_{t=1}^{\iota} \log P(\tau_t | \tau_{<t}, S). \quad (4)$$

In this way, we exert explicit supervision to encourage the model to condition on the input personas.

3.2 Plot Planning

At this stage, CONPER learns to plan a keyword sequence for subsequent story generation (Yao et al., 2019). Plot planning requires a strong ability to model the causal and temporal relationship in the context for expanding a reasonable story plot (e.g., associating “unskilled” with “failure” for the example in Table 1), which is extremely challenging without any external guidance, for instance, commonsense knowledge. In order to plan a coherent event sequence, we introduce a dynamically growing local knowledge graph, a subset of the external commonsense knowledge base ConceptNet (Speer et al., 2017), which is initialized to contain triples related to the keywords mentioned in the input and target. When planning the next keyword, CONPER combines the knowledge information from the local graph and the contextualized features captured by the language model with learnable weights. Then CONPER grows the local graph by adding the knowledge triples neighboring the predicted keyword. Formally, we denote the keyword sequence as $W = (w_1, w_2, \dots, w_k)$ of length k and the local graph as \mathcal{G}_t for predicting the keyword w_t . The loss function \mathcal{L}_{KW} for generating the keyword sequence is as follows:

$$\mathcal{L}_{KW} = - \sum_{t=1}^k \log P(w_t | w_{<t}, S, T, \mathcal{G}_t). \quad (5)$$

Keyword Extraction We extract words that relate to emotions and events from each sentence of a story as keywords for training, since they are important for modeling characters’ evolving psychological states and their behavior. We measure

the emotional tendency of each word using the sentiment analyzer in NLTK, which predicts a distribution over four basic emotions, i.e., *negative*, *neutral*, *positive*, and *compound*. We regard those words as related to emotions whose scores for *negative* or *positive* are larger than 0.5. Secondly, we extract and lemmatize the nouns and verbs (excluding stopwords) from a story as event-related keywords with NLTK for POS-tagging and lemmatization. Then we combine the two types of keywords in the original order as the keyword sequence for planning. We limit the number of keywords extracted from each sentence in stories up to 5, and we ensure that there is at least one keyword for a sentence by randomly choosing one word if no keywords are extracted. We don’t keep this limitation when extracting keywords from the leading context and the persona description, since these keywords are only used to initialize the local knowledge graph.

Incorporating Knowledge We introduce a dynamically growing local knowledge graph for plot planning. For each example, we initialize the graph \mathcal{G}_1 as a set of knowledge triples where the keywords in S and T are the head or tail entities, and then update \mathcal{G}_t to \mathcal{G}_{t+1} by adding triples related with the generated keyword w_t at t -th step. Then, the key problem at this stage is representing and utilizing the local graph for next keyword prediction.

The local graph consists of multiple sub-graphs, each of which contains all the triples related with a keyword denoted as $\varepsilon_i = \{(h_n^i, r_n^i, t_n^i) | h_n^i \in \mathcal{V}, r_n^i \in \mathcal{R}, t_n^i \in \mathcal{V}\}_{n=1}^N$, where \mathcal{R} and \mathcal{V} are the relation set and entity set of ConceptNet, respectively. We derive the representation \mathbf{g}_i for ε_i using graph attention (Zhou et al., 2018b) as follows:

$$\mathbf{g}_i = \sum_{n=1}^N \alpha_n [\mathbf{h}_n^i; \mathbf{t}_n^i] \quad (6)$$

$$\alpha_n = \frac{\exp(\beta_n)}{\sum_{j=1}^N \exp(\beta_j)}, \quad (7)$$

$$\beta_n = (\mathbf{W}_r \mathbf{r}_n^i)^T \tanh(\mathbf{W}_h \mathbf{h}_n^i + \mathbf{W}_t \mathbf{t}_n^i), \quad (8)$$

where \mathbf{W}_h , \mathbf{W}_r and \mathbf{W}_t are trainable parameters, \mathbf{h}_n^i , \mathbf{r}_n^i and \mathbf{t}_n^i are learnable embedding representations for h_n^i , r_n^i and t_n^i , respectively. We use the same BPE tokenizer (Radford et al., 2019) with the language model to tokenize the head and tail entities, which may lead to multiple sub-words for an entity.

Therefore, we derive \mathbf{h}_n^i and \mathbf{t}_n^i by adding the

embeddings of all the sub-words. And we initialize the relation embeddings randomly.

After obtaining the graph representation, we predict the distribution of the next keyword by dynamically deciding whether to select the keyword from the local graph as follows:

$$P(w_t|w_{<t}, S, T, \mathcal{G}_t) = \gamma_t P_k^t + (1 - \gamma_t) P_l^t, \quad (9)$$

where $\gamma_t \in \{0, 1\}$ is a binary learnable weight, and P_l^t is a distribution over the whole vocabulary while P_k^t is a distribution over the entities in \mathcal{G}_t . We incorporate the knowledge information implicitly for computing both distributions:

$$P_k^t = \text{softmax}(\mathbf{W}_k[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_k), \quad (10)$$

$$P_l^t = \text{softmax}(\mathbf{W}_l[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_l), \quad (11)$$

where $\mathbf{W}_k, \mathbf{b}_k, \mathbf{W}_p$ and \mathbf{b}_p are trainable parameters, and \mathbf{c}_t is a summary vector of the knowledge information by attending on the representations of all the sub-graphs in \mathcal{G}_t , formally as follows:

$$\mathbf{c}_t = \sum_{n=1}^N \alpha_n \mathbf{g}_n, \quad (12)$$

$$\alpha_n = \text{softmax}(\mathbf{s}_t^T \mathbf{W}_g \mathbf{g}_n). \quad (13)$$

where \mathbf{W}_g is a trainable parameter. During training process, we set γ_t to the ground-truth label $\hat{\gamma}_t$. During generation process, we decide γ_t by deriving the probability p_t of selecting an entity from the local graph as the next keyword. And we set γ_t to 1 if $p_t < 0.5$ otherwise 0. We compute p_t as follows:

$$p_t = \text{sigmoid}(\mathbf{W}_p[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_p), \quad (14)$$

where \mathbf{W}_p and \mathbf{b}_p are trainable parameters. We train the classifier with the standard cross entropy loss \mathcal{L}_C derived as follows:

$$\mathcal{L}_C = -(\hat{\gamma}_t \log p_t + (1 - \hat{\gamma}_t) \log(1 - p_t)), \quad (15)$$

where $\hat{\gamma}_t$ is the ground-truth label. In summary, the overall loss function \mathcal{L}_{PP} for the plot planning stage is computed as follows:

$$\mathcal{L}_{PP} = \mathcal{L}_{KW} + \mathcal{L}_C. \quad (16)$$

By incorporating commonsense knowledge for planning, and dynamically updating the local graph, CONPER can better model the causal and temporal relationship between events in the context.

Target Guidance In order to further improve the coherence and the persona-consistency, we propose to exert explicit guidance of the predicted target on plot planning. Specifically, we expect CONPER to predict keywords close to the target in semantics. Therefore, we add a bias term \mathbf{d}_k^t and \mathbf{d}_l^t into Equation 10 and 11, respectively, formally as follows:

$$P_k^t = \text{softmax}(\mathbf{W}_k[\mathbf{s}_t; \mathbf{c}_t] + \mathbf{b}_k + \mathbf{d}_k^t), \quad (17)$$

$$\mathbf{d}_k^t = [\mathbf{s}_{\text{tar}}; \mathbf{c}_t]^T \mathbf{W}_d \mathbf{E}_k + \mathbf{b}_d, \quad (18)$$

$$\mathbf{s}_{\text{tar}} = \frac{1}{l} \sum_{t=1}^l \mathbf{s}_{|\tau_t|}, \quad (19)$$

where \mathbf{W}_d and \mathbf{b}_d are trainable parameters, \mathbf{s}_{tar} is the target representation computed by averaging the hidden states at each position of the predicted target, and \mathbf{E}_k is an embedding matrix, each row of which is the embedding for an entity in \mathcal{G}_t . The modification for Equation 11 is similar except that we compute the bias term \mathbf{d}_l^t with an embedding matrix \mathbf{E}_l for the whole vocabulary.

3.3 Story Generation

After planning the target T and the keyword sequence W , we train CONPER to generate the whole story conditioned on the input and plans with the standard language model loss \mathcal{L}_{ST} . Since we extract one sentence from a story as the target, we do not train CONPER to regenerate the sentence in the story generation stage. And we insert a special token `Target` in the story to specify the position of the target during training. In the inference time, CONPER first plans the target and plot, then generates the whole story, and finally places the target into the position of `Target`.

4 Experiments

4.1 Dataset

We conduct the experiments on the STORIUM dataset (Akoury et al., 2020). STORIUM contains nearly 6k long-form stories and each story unfolds through a series of scenes with several shared characters. A scene consists of multiple short scene entries, each of which is written to either portray one character with annotation for his personality (i.e., the “card” in STORIUM), or introduce new story settings (e.g., problems, locations) from the perspective of the narrator. In this paper, we concatenate all entries from the same scene since a scene can be seen as an independent story. And we regard a scene entry written for a certain character as the

target output, the personality of the character as the persona description, and the previous entries written for this character or from the perspective of the narrator in the same scene as the leading context. We split the processed examples for training, validation and testing based on the official split of STORIUM. We retain about 1,000 words (with the correct sentence boundary) for each example due to the length limit of the pretrained language model.

At the plot planning stage, we retrieve a set of triples from ConceptNet (Speer et al., 2017) for each keyword extracted from the input or generated by the model. We only retain those triples of which both the head and tail entity contain one word and occur in our dataset, and the confidence score of the relation (annotated by ConceptNet) is more than 1.0. The average number of triples for each keyword is 33. We show more statistics in Table 2.

	Train	Valid	Test
# Examples	47,910	6,477	6,063
Avg. Context Length	332.7	324.8	325.7
Avg. Description Length	23.8	22.7	24.6
Avg. Story Length	230.5	225.7	234.3
Avg. Target Length	21.8	21.7	22.2
Avg. # Keywords (Input)	101.1	99.2	99.6
Avg. # Keywords (Story)	31.2	30.5	31.5

Table 2: Dataset statistics. We compute the length by counting tokens using the BPE tokenizer of GPT2. Keywords are extracted either from the input to initialize the local graph, or from the story to train the model for plot planning.

4.2 Baselines

We compare CONPER with following baselines. (1) **ConvS2S**: It directly uses a convolutional seq2seq model to generate a story conditioned on the input (Gehring et al., 2017). (2) **Fusion**: It generates a story by first training a convolutional seq2seq model, and then fixing the model and initializing another trainable convolutional seq2seq model with its parameters. Then the two models are trained together by a fusion mechanism. (Fan et al., 2018). (3) **Plan&Write**: It first plans a keyword sequence conditioned on the input, and then generates a story based on the keywords (Yao et al., 2019). (4) **GPT2_{Scr}**: It has the same network architecture with GPT2 but is trained on our dataset from scratch without any pretrained parameters. (5) **GPT2_{Ft}**: It is initialized using pretrained parameters, and then fine-tuned on our dataset with the standard language modeling objective. (6) **PlanA-**

head: It first predicts a keyword distribution conditioned upon the input, and then generates a story by combining the language model prediction and the keyword distribution with a gate mechanism (Kang and Hovy, 2020). We remove the sentence position embedding and the auxiliary training objective (next sentence prediction) used in the original paper for fair comparison.

Furthermore, we evaluate the following ablated models to investigate the influence of each component: (1) **CONPER w/o KG**: removing the guidance of the commonsense knowledge in the plot planning stage. (2) **CONPER w/o TG**: removing target guidance in the plot planning stage. (3) **CONPER w/o PP**: removing the plot planning stage, which means the model first plans a target sentence and then directly generates the whole story. (4) **CONPER w/o TP**: removing the target planning stage, which also leads to the removal of target guidance in the plot planning stage.

4.3 Experiment Settings

We build CONPER based on GPT2 (Radford et al., 2019), which is widely used for story generation (Guan et al., 2020). We concatenate the context and the persona description with a special token as input for each example. For fair comparison, we also add special tokens at both ends of the target sentence in a training example for all baselines. We implement the non-pretrained models based on the scripts provided by the original papers, and the pretrained models based on the public checkpoints and codes of HuggingFace’s Transformers*. And we set all the pretrained models to the base version due to limited computational resources. We set the batch size to 8, the initial learning rate of the AdamW optimizer to 5e-5, and the maximum training epoch to 5 with an early stopping mechanism. And we generate stories using top- p sampling with $p = 0.9$ (Holtzman et al., 2019). We apply these settings to all the GPT-based models, including GPT_{Scr}, GPT_{Ft}, PlanAhead, CONPER and its ablated models. As for ConvS2S, Fusion and Plan&Write, we used the settings from their respective papers and codebases.

4.4 Automatic Evaluation

Metrics We adopt the following automatic metrics for evaluation on the test set. (1) **BLEU (B-n)**: We use $n = 1, 2$ to evaluate n -gram overlap

*<https://github.com/huggingface/transformers>

Models	Coherence				Persona consistency			
	Win(%)	Lose(%)	Tie(%)	κ	Win(%)	Lose(%)	Tie(%)	κ
CONPER vs. ConvS2S	89.0*	5.0	6.0	0.625	82.0*	8.0	10.0	0.564
CONPER vs. Fusion	71.0*	23.0	6.0	0.213	61.0*	22.0	17.0	0.279
CONPER vs. GPT2 _{Ft}	54.0*	18.0	28.0	0.275	53.0*	11.0	36.0	0.215
CONPER vs. PlanAhead	53.0*	25.0	22.0	0.311	59.0*	28.0	13.0	0.280

Table 3: Manual evaluation results. The scores indicate the percentage of *win*, *lose* or *tie* when comparing our model with a baseline. κ denotes Randolph’s kappa to measure the inter-annotator agreement. * means CONPER outperforms the baseline model significantly with p -value < 0.01 (Wilcoxon signed-rank test).

Models	B-1	B-2	BS-t	BS-m	PC
ConvS2S	12.5	4.7	22.2	32.8	17.1
Fusion	13.3	5.0	22.7	33.3	30.8
Plan&Write	7.2	2.8	6.2	29.7	23.6
GPT2 _{Scr}	13.3	4.8	24.7	38.0	26.6
GPT2 _{Ft}	13.5	4.7	26.7	37.8	39.5
PlanAhead	15.4	5.3	26.1	37.8	50.2
CONPER	19.1	6.9	32.1	41.4	59.7
w/o KG	17.4	<u>6.3</u>	31.6	39.7	53.4
w/o TG	<u>17.7</u>	<u>6.3</u>	31.9	<u>40.2</u>	<u>56.3</u>
w/o PP	14.9	5.3	<u>32.0</u>	40.0	46.9
w/o TP	16.4	5.8	<u>27.8</u>	37.7	44.9
Grouth Truth	N/A	N/A	42.6	42.6	75.2

Table 4: Automatic evaluation results. The best performance is highlighted in **bold**, and the second best is underlined. All results are multiplied by 100.

between generated and ground-truth stories (Papineni et al., 2002). (2) **BERTScore-target (BS-t)**: We use $\text{BERTScore}_{\text{Recall}}$ (Zhang et al., 2019) to measure the semantic similarity between the generated target sentence and the persona description. A higher result indicates the target embodies the persona better. (3) **BERTScore-max (BS-m)**: It computes the maximum value of BERTScore between each sentence in the generated story and the persona description. (4) **Persona-Consistency (PC)**: It is a learnable automatic metric (Guan and Huang, 2020). We fine-tune $\text{RoBERTa}_{\text{BASE}}$ on the training set as a classifier to distinguish whether a story exhibits a consistent persona with a persona description. We regard the ground-truth stories as positive examples where the stories and the descriptions are consistent, and construct negative examples by replacing the story with a randomly sampled one. After fine-tuning, the classifier achieves an 83.63% accuracy on the auto-constructed test set. Then we calculate the consistency score as the average classifier score of all the generated texts regarding the corresponding input.

Result Table 4 shows the automatic evaluation results. CONPER can generate more word overlaps with ground-truth stories as shown by higher BLEU scores. And CONPER can better embody the specified persona in the target sentence and the whole story as shown by the higher BS-t and BS-m score. The higher PC score of CONPER also further demonstrate the better exhibition of given personas in the generated stories. As for ablation tests, all the ablated models have lower scores in terms of all metrics than CONPER, indicating the effectiveness of each component. Both CONPER w/o PP and CONPER w/o TP drop significantly in BLEU scores, suggesting that planning is important for generating long-form stories. CONPER w/o TP also performs substantially worse in all metrics than CONPER w/o TG, indicating the necessity of explicitly modeling the relations between persona descriptions and story plots. We also show analysis of target guidance in Appendix C.

4.5 Manual Evaluation

We conduct a pairwise comparison between our model and four strong baselines including PlanAhead, GPT2_{Ft}, Fusion and ConvS2S. We randomly sample 100 stories from the test set, and obtain 500 stories generated by CONPER and four baseline models. For each pair of stories (one by CONPER, and the other by a baseline, along with the input), we hire three annotators to give a preference (*win*, *lose* or *tie*) in terms of *coherence* (inter-sentence relatedness, causal and temporal dependencies) and *persona-consistency* with the input (exhibiting consistent personas). We adopt majority voting to make the final decisions among three annotators. Note that the two aspects are independently evaluated. We resort to Amazon Mechanical Turk (AMT) for the annotation. As shown in Table 3, CONPER outperforms baselines significantly in coherence and persona consistency.

Furthermore, we used human annotation to eval-

Policies	Yes (%)	No (%)	κ
Random	22.0	78.0	0.25
Ours	75.0	25.0	0.36

Table 5: Percentages of examples labeled with “Yes” or “No” for whether the identified sentence reflects the given persona. κ denotes Randolph’s kappa to measure the inter-annotator agreement.

uate whether the identified target sentence embodies the given persona. We randomly sampled 100 examples from the test set, and identified the target for each example as the sentence with the maximum BERTScore with the persona description. And we used a random policy as a baseline which randomly samples a sentence from the original story as the target. We hired three annotators on AMT to annotate each example (“Yes” if the sentence embodies the given persona, and “No” otherwise). We adopted majority voting to make the final decision among three annotators. Table 5 shows our method significantly outperforms the random policy in identifying the persona-related sentences.

4.6 Controllability Analysis

To further investigate whether the models can be generalized to generate specific stories to exhibit different personas conditioned on the same context, we perform a quantitative study to observe how many generated stories are successfully controlled as the input persona descriptions change.

Automatic Evaluation For each example in the test set, we use a model to generate ten stories conditioned on the context of this example and ten persona descriptions randomly sampled from other examples, respectively. We regard a generated story as successfully controlled if the pair of the story and its corresponding persona description (along with the context) has the maximum persona-consistency score among all the ten descriptions. We regard the average percentages of the stories which are successfully controlled in all the ten generated stories for each example in the whole test set as the controllability score of the model. We show the results for CONPER and strong baselines in Table 6. Furthermore, we also compute the superiority (denoted as Δ) of the persona-consistency score computed between a generated story and its corresponding description compared to that computed between the story and one of the other nine descriptions (Sinha et al., 2020). A larger Δ means the model can gen-

Models	Controllability Score	Δ
Plan&Write	10.6	0.01
GPT2 _{Ft}	24.2	11.2
PlanAhead	23.1	11.2
CONPER	29.5	15.1

Table 6: Automatic evaluation results for the controllability. All results are multiplied by 100.

Models	Acco (%)	Oppo (%)	Irre (%)
GPT2 _{Ft}	21	10	69
PlanAhead	44	12	44
CONPER	66	9	25

Table 7: Manual evaluation results for the controllability. Acco/Oppo/Irre means the example exhibits an accordant/opposite/irrelevant persona with the input.

erate more specific stories adhering to the personas.

As shown in Table 6, there are more stories successfully controlled for CONPER than baselines. And the larger Δ of CONPER suggests that it can generate more specific stories to the input personas. The results show the better generalization ability of CONPER to generate persona-controllable stories.

Manual Evaluation For manual evaluation, we randomly sampled 50 examples from the test set, and manually revised the persona descriptions to exhibit an opposite persona (e.g., from “skilled pilot” to “unskilled pilot”). We required a model to generate two stories conditioned on the original and its opposite persona description, respectively. Finally we obtained 300 stories from three models including GPT2_{Ft}, PlanAhead and CONPER. Then, we hired three graduates to judge whether each story accords with the input persona. All annotators have good English language proficiency and are well trained for this evaluation task. Table 7 shows the evaluation results. We can see that 66% of the stories generated by CONPER are accordant with the input persona, suggesting the better controllability of CONPER.

4.7 Case Study

We present some cases in Table 8. We can see that the story generated by CONPER exhibits the specified persona with a coherent event sequence. The planned keywords by CONPER provide an effective discourse-level guidance for the subsequent story generation, such as `tablet`, which has a common-sense connection with `computer skills` and

Context: ... the group has gathered on the rooftop garden of Miyamoto Mansion ... the TV set out near the long table on the patio is talking about some spree of thefts at low volume ... the issue of Chloe’s disappearance and the missing statue still hang over their heads.

Persona Description: [Aito] You are above average in your computer skills. If information is power, then your ability to use the internet makes you one of the most powerful people on the planet.

GPT2_{FT}: *Aito looked at the others, still trying to help find a way out of the hotel. He wasn’t sure what the rest of the group wanted to see if they were going to survive and all knew if he needed to be needed ...*

PlanAhead: Miyamoto Mansion ... perhaps it’s just a bit farther away. The music sounds bright enough but the line of visitors does not. *Aito was once a pretty girl, he had always been quite witty when talking to people but she always found it annoying that a group of tourists looked like trash just to her ...*

CONPER: ... “Oh, wait ... wait ... people are talking about Chloe?” ... **“I have a feeling the internet is probably our best chance to get through this”** ... Aito looked around the table a moment before pulling out her tablet and starting typing furiously into her computer. She looked up at the tablet that had appeared, and she could see that it was working on a number of things ...

Planned keywords: ... people → look → around → tablet → see ...

Table 8: Generated stories by different models. *Italic* words indicate the improper entities or events in terms of the consistency with the input. The **bold** sentence indicate the generated target by CONPER. **Red** words denote the consistent events adhering to the input. And the extracted keywords are underlined.

Internet in the input. In contrast, the baselines tend to not generate any persona-related events. For example, the given persona description emphasizes the strong computer skills of the protagonist while the stories generated by PlanAhead and GPT2 have nothing to do with the computer skills. We further analyze some error cases generated by our model in Appendix G.

5 Conclusion

We present CONPER, a planning-based model for a new task aiming at controlling the protagonist’s persona in story generation. We propose target planning to explicitly model the relations between persona-related events and input personas, and plot planning to learn the keyword transition in a story with the guidance of predicted persona-related events and external commonsense knowledge. Extensive experiments show that CONPER can generate more coherent stories with better consistency with the input personas than strong baselines. Further analysis also indicates the better persona-controllability of CONPER.

Acknowledgement

This work was supported by the National Science Foundation for Distinguished Young Scholars (with No. 62125604) and the NSFC projects (Key project with No. 61936010 and regular project with No. 61876096). This work was also supported by the Guoqiang Institute of Tsinghua University, with Grant No. 2019GQG1 and 2020GQG0005. This

work was also sponsored by Tsinghua-Toyota Joint Research Fund. We would also like to thank the anonymous reviewers for their invaluable suggestions and feedback.

Ethics Statements

We conduct the experiments by adapting a public story generation dataset STORIUM to our task. Automatic and manual evaluation results show that our model CONPER outperforms existing state-of-the-art models in terms of coherence, consistency and controllability, suggesting the generalization ability of CONPER to different input personas. And our approach can be easily extended to different syntactic levels (e.g., phrase-level and paragraph-level events), different model architectures (e.g., BART (Lewis et al., 2020)) and different generation tasks (e.g., stylized long text generation).

In both STORIUM and ConceptNet, we find some potentially offensive words. Therefore, our model may suffer from risks of generating offensive content, although we have not observed such content in the generated results. Furthermore, ConceptNet consists of commonsense triples of concepts, which may not be enough for modeling inter-event relations in long-form stories. We resort to Amazon Mechanical Turk (AMT) for manual evaluation. We do not ask about personal privacy or collect personal information of annotators in the annotation process. We hire three annotators and pay each annotator \$0.1 for comparing each pair of stories. The payment is reasonable considering that it

would cost average one minute for an annotator to finish a comparison.

References

- Nader Akoury, Shufan Wang, Josh Whiting, Stephen Hood, Nanyun Peng, and Mohit Iyyer. 2020. Storium: A dataset and evaluation platform for machine-in-the-loop story generation. *arXiv preprint arXiv:2010.01717*.
- Prithviraj Ammanabrolu, Wesley Cheung, William Broniec, and Mark O Riedl. 2021. Automated storytelling via causal, commonsense plot ordering. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 5859–5867.
- David Bamman, Brendan O’Connor, and Noah A Smith. 2013. Learning latent personas of film characters. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 352–361.
- David Bamman, Ted Underwood, and Noah A Smith. 2014. A bayesian mixed effects model of literary character. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 370–379.
- Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. "O’Reilly Media, Inc."
- Faeze Brahman and Snigdha Chaturvedi. 2020. Modeling protagonist emotions for emotion-aware storytelling. *arXiv preprint arXiv:2010.06822*.
- Faeze Brahman, Meng Huang, Oyvind Tafjord, Chao Zhao, Mrinmaya Sachan, and Snigdha Chaturvedi. 2021. "let your characters tell their story": A dataset for character-centric narrative understanding. *arXiv preprint arXiv:2109.05438*.
- Faeze Brahman, Alexandru Petrusca, and Snigdha Chaturvedi. 2020. Cue me in: Content-inducing approaches to interactive story generation. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 588–597.
- Marc Cavazza, David Pizzi, Fred Charles, Thurid Vogt, and Elisabeth André. 2009. Emotional input for character-based interactive storytelling. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems-Volume 1*, pages 313–320.
- 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.
- Elizabeth Clark, Yangfeng Ji, and Noah A Smith. 2018. Neural text generation in stories using entity representations as context. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2250–2260.
- 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, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. *arXiv preprint arXiv:1805.04833*.
- Angela Fan, Mike Lewis, and Yann Dauphin. 2019. Strategies for structuring story generation. *arXiv preprint arXiv:1902.01109*.
- Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In *International Conference on Machine Learning*, pages 1243–1252. PMLR.
- Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph M. Weischedel, and Nanyun Peng. 2020. Content planning for neural story generation with aristotelian rescoring. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 4319–4338. Association for Computational Linguistics.
- Arthur C Graesser, Kathy L Lang, and Richard M Roberts. 1991. Question answering in the context of stories. *Journal of Experimental Psychology: General*, 120(3):254.
- Jian Guan, Fei Huang, Zhihao Zhao, Xiaoyan Zhu, and Minlie Huang. 2020. A knowledge-enhanced pre-training model for commonsense story generation. *Transactions of the Association for Computational Linguistics*, 8:93–108.
- Jian Guan and Minlie Huang. 2020. UNION: an un-referenced metric for evaluating open-ended story generation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 9157–9166. Association for Computational Linguistics.
- Jian Guan, Yansen Wang, and Minlie Huang. 2019. Story ending generation with incremental encoding and commonsense knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6473–6480.
- Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2019. The curious case of neural text degeneration. *arXiv preprint arXiv:1904.09751*.

- Yangfeng Ji, Chenhao Tan, Sebastian Martschat, Yejin Choi, and Noah A Smith. 2017. Dynamic entity representations in neural language models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1830–1839.
- Dongyeop Kang and Eduard Hovy. 2020. Plan ahead: Self-supervised text planning for paragraph completion task. *arXiv preprint arXiv:2010.05141*.
- 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 (Volume 1: Long Papers)*, 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. *CoRR*, abs/1909.05858.
- 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, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.
- Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and William B Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of 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, Barcelona, Spain. Association for Computational Linguistics.
- Zibo Lin, Deng Cai, Yan Wang, Xiaojiang Liu, Hai-Tao Zheng, and Shuming Shi. 2020. The world is not binary: Learning to rank with grayscale data for dialogue response selection. *arXiv preprint arXiv:2004.02421*.
- Danyang Liu, Juntao Li, Meng-Hsuan Yu, Ziming Huang, Gongshen Liu, Dongyan Zhao, and Rui Yan. 2020. A character-centric neural model for automated story generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 1725–1732.
- Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.
- Sabine Madsen and Lene Nielsen. 2009. Exploring persona-scenarios-using storytelling to create design ideas. In *IFIP Working Conference on Human Work Interaction Design*, pages 57–66. Springer.
- Huanru Henry Mao, Bodhisattwa Prasad Majumder, Julian McAuley, and Garrison W Cottrell. 2019. Improving neural story generation by targeted common sense grounding. *arXiv preprint arXiv:1908.09451*.
- Tong Niu and Mohit Bansal. 2018. Polite dialogue generation without parallel data. *Transactions of the Association for Computational Linguistics*, 6:373–389.
- 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, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.
- Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. *arXiv preprint arXiv:2004.14967*.
- Mark O Riedl and Robert Michael Young. 2010. Narrative planning: Balancing plot and character. *Journal of Artificial Intelligence Research*, 39:217–268.
- Koustuv Sinha, Prasanna Parthasarathi, Jasmine Wang, Ryan Lowe, William L. Hamilton, and Joelle Pineau. 2020. Learning an unreferenced metric for online dialogue evaluation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2430–2441, Online. Association for Computational Linguistics.
- Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. Conceptnet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 31.
- Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric P Xing, and Zhiting Hu. 2019. Target-guided open-domain conversation. *arXiv preprint arXiv:1905.11553*.
- 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, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 5233–5239. ijcai.org.
- 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, EMNLP 2020, Online*,

November 16-20, 2020, pages 2831–2845. Association for Computational Linguistics.

Pengcheng Yang, Lei Li, Fuli Luo, Tianyu Liu, and Xu Sun. 2019. [Enhancing topic-to-essay generation with external commonsense knowledge](#). In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 2002–2012. Association for Computational Linguistics.

Lili Yao, Nanyun Peng, Ralph Weischedel, Kevin Knight, Dongyan Zhao, and Rui Yan. 2019. Plan-and-write: Towards better automatic storytelling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 7378–7385.

R Michael Young. 2000. Creating interactive narrative structures: The potential for ai approaches. *Psychology*, 13:1–26.

Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too? *arXiv preprint arXiv:1801.07243*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. Bertscore: Evaluating text generation with bert. *arXiv preprint arXiv:1904.09675*.

Yinhe Zheng, Rongsheng Zhang, Minlie Huang, and Xiaoxi Mao. 2020. A pre-training based personalized dialogue generation model with persona-sparse data. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9693–9700.

Peixiang Zhong, Yong Liu, Hao Wang, and Chunyan Miao. 2020. [Keyword-guided neural conversational model](#). *CoRR*, abs/2012.08383.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018a. [Commonsense knowledge aware conversation generation with graph attention](#). In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI 2018, July 13-19, 2018, Stockholm, Sweden*, pages 4623–4629. ijcai.org.

Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018b. Commonsense knowledge aware conversation generation with graph attention. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4623–4629.

A Implementation Details

We train our model on one Quadro RTX 6000 GPU. It costs about 25 hours to train our model, and 4 hours to generate stories using our model.

Models	B-1	B-2	BS-t	BS-m	PC
CONPER	19.1	6.9	32.1	41.4	59.7
CONPER (Rand)	17.4	6.2	26.0	38.9	52.1
CONPER (Multi)	17.9	6.6	32.6	40.0	55.1
CONPER (Sparse)	18.0	6.6	31.6	40.2	57.0
<i>Growth Truth</i>	<i>N/A</i>	<i>N/A</i>	42.6	42.6	75.2

Table 9: Automatic evaluation results for several variants of CONPER. The best performance is highlighted in **bold**. All results are multiplied by 100.

B Analysis of Extraction Strategy

B.1 Target Extraction

We regard one sentence which has the maximum BERTScore with the persona description as the target in our model. We conducted two experiments to further investigate the influence of target extraction strategy: **(1) CONPER (Rand)**: It regards a sentence randomly sampled from the story as the target for training in the target planning stage. **(2) CONPER (Multi)**: It regards two sentences which have the maximum BERTScore with the persona description as the target.

As shown in Table 9, when using a random sentence as the target, all the metrics drop significantly. And Table 5 in the main paper shows that it is hard for the random policy to select persona-related sentences. The results indicate the benefit of our methods for modeling relations between personas and events. Moreover, using multiple sentences as the target is inferior to using only one in terms of most metrics. It is possibly because stories in STORIUM tend to embody personas sparsely, and modeling the relations between personas and multiple persona-unrelated events directly may hurt the performance. The BS-t score is higher when using multiple sentences because more words can easily lead to a higher recall score.

B.2 Keyword Extraction

We extracted at most 5 keywords from each sentence for the plot planning stage. We also experimented with a more sparse plan by extracting only one keyword from each sentence (called **CONPER (Sparse)**). Table 9 shows that using a more sparse plan performs worse in all metrics. It is possibly because the limited planning keywords could not make the best of the external knowledge to form coherent and persona-related plots.

C Analysis of Target Guidance

We visualize how target guidance affects word prediction in the plot planning stage in Figure 2. The original word distribution is weighted to those words irrelevant to the target sentence, while the bias term (Equation 18) is weighted to those words related to the target sentence in semantics such as *bar*. After combining the original word distribution with the bias term, the final distribution can balance the trade-off between target guidance and language model prediction. This validates our hypothesis that target guidance can draw the planned plots closer to the target, which helps improve the story coherence and persona-consistency.

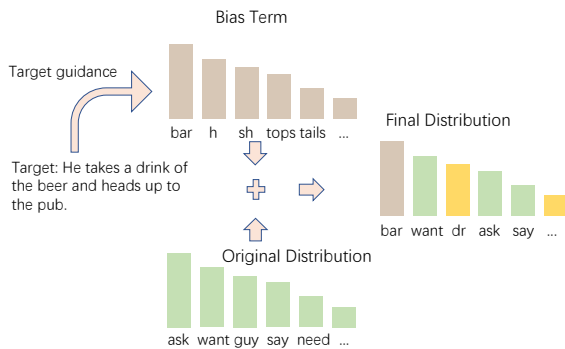


Figure 2: A case showing the effect of target guidance. The planning keywords are brought closer to the target in semantics under the target guidance.

D Diversity

We compare the diversity of CONPER with baselines using distinct- n (D- n) (Li et al., 2016), the ratio of distinct n -grams to all n -grams in generated stories. The results in Table 10 show that CONPER has better coherence and persona consistency without sacrificing the diversity.

Models	D-1	D-2	D-3	D-4
GPT2 _{Scr}	0.021	0.134	0.381	0.653
GPT2 _{Ft}	0.022	0.184	0.501	0.777
PlanAhead	0.032	0.256	0.618	0.863
CONPER	0.016	0.148	0.439	0.730
<i>Growth Truth</i>	<i>0.062</i>	<i>0.368</i>	<i>0.739</i>	<i>0.927</i>

Table 10: Automatic evaluation results. CONPER is comparable with fine-tuned GPT2 in diversity performance.

Models	Number of Parameters
ConvS2S	135M
Fusion	255M
GPT2	124M
PlanAhead	201M
CONPER	247M

Table 11: Number of Parameters of different models.

E Manual Evaluation

We conduct manual evaluation on Amazon Mechanical Turk. To improve the annotation quality, we provide a detailed instruction for annotators, which contains: (1) a summary of our task; (2) a formal definition for coherence and persona consistency; and (3) good and bad examples for coherence and persona consistency. The detailed evaluation guideline is shown in Figure 3.

F Model Parameters

We compute the number of parameters for some models used in our experiments. The result is shown in Table 11.

G Error Analysis

Although the proposed model outperforms the strong baselines, Table 7 in the main paper shows that there are still many generated stories that exhibit opposite or irrelevant with the given persona. Therefore, we presented some typical error cases generated by our model for each error type in Figure 4. These cases show our model still does not completely control personas in story generation. When there is a slight conflict between the generated target sentence and the given persona (e.g., *you're here for fun* is slightly conflict with *slow to action*), the generated plan would further deviate from the input under the guidance of the target sentence (e.g., *excit, like*), and finally the generated story exhibits an opposite persona. Similarly, when the generated target sentence is irrelevant with the given persona (e.g., *That was the hardest thing too see*), the final generated story doesn't have any persona-related event. These errors also indicate the target sentence plays an important role in controlling the protagonist's persona in story generation.

Persona	Generated Stories	PC	Rouge-2
You have a loose grasp on your emotion and are quick to lash out. Often hurting the ones you care about most.	... She had started to try to get her emotions under control, she had tried to keep herself calm during all of this, and she had only gotten worse. She stood, just standing, and was staring at the class like an idiot, her head down...	0.90	0.0
<u>In command of</u> some measure of magical power	...Wilhelm is <u>in command of</u> all forces in the north. After the war ended, his mind was full of food and drink, and he was ready for a quick trip to the bar. He had been to the bar on many occasions. Only in there that he could fully relax and forget about all those mess things...	0.16	0.14

Table 12: Typical cases by PC (Persona-Consistency) score. **Bold** words denote the consistent events adhering to the given persona. The overlapping words are underlined.

H Discussion of the Persona-Consistency Metric

To measure whether the generated story is consistent with the given persona, we propose the Persona-Consistency(PC) metric. In our experiments, we replace the ground-truth story with a randomly sampled one to construct a inconsistent story-persona pair as a negative sample. The fine-tuned classifier achieves an 83.63% accuracy on the auto-constructed test set. However, it is possible that the PC metric depends on the word overlap to make predictions because of the simple random sampling of the negative samples (Lin et al., 2020). We thus conduct a case study to investigate whether our PC metric depends on word overlap to make judgments. As shown in Table 12, the first example gets a high PC score since the story embodies a consistent persona with the given persona description, in spite of a low rouge score. In contrast, the second example has an overlapped phrase “in command of” with the persona description but does not embody the corresponding persona description, and thus gets a high rouge score and low PC score. The results show that PC may not depend on shallow features like word overlap to make judgments.

What’s more, we have taken into account the shortcomings of the automatic metrics for NLG and thus additionally added the human evaluation to further prove the effectiveness of our method.

Human Evaluation Guideline	
<p>This study aims to evaluate automatic story generation systems. Specifically, for each story, we will put the context and the persona description into a generative system, and the following sentences will be generated by the system. In the process of evaluation, you will be given two generated stories by two difference systems, denoted as A, B. You need to compare A with B in terms of the coherence and persona consistency.</p> <p>Notes:</p> <ul style="list-style-type: none"> ◆ Please annotate the stories carefully based on comprehensive comparison and technically following the instruction. ◆ Please make a more fine-grained comparison when annotating persona consistency, as it is very common that neither of the two systems can clearly reflect the given persona. Be extra careful when you choose tie. 	
<p>➤ Coherence: context relatedness and inter-sentence causal and temporal dependencies.</p>	
Options	1. System A is better 2. System B is better 3. System A and System B are good or bad equally.
e.g.	<p>Persona: [Pintoos] Go where you've never been before and share your experiences through song and story.</p> <p>Context: You wake up from an uneasy rest in the Inn. The rain batters the window and occasional lightning briefly illuminates the tops of low-rising huts and hovels outside. As your level of consciousness increases you become vaguely aware of a commotion of some sort coming from outside your door...</p> <p>Generated Story 1: Pintoos descends the wooden stairway from his shabby <u>inn-room</u> confident that this day will be the beginning of his journey to new lands, interesting characters, and forgotten knowledge. Approaching the <u>innkeeper</u> he sits and surveys the room. There is a <u>steady dripping of rainwater falling from a leak in the inn's roof</u>, a few inches from his head... "Say, what brings you here all the way from... well where you from stranger?" Pintoos nods, "Yes, my name is Pintoos I've just come here from the Southern rains. D'welsee is what we call our homeland. There once a boy is of age he is given leave of the land to explore the world and learn... well anything. (Good coherence, some premise related parts are underlined)</p> <p>Generated Story 2: Something about this commotion is... unsettling, and it leaves his mind racing. Well, I should probably <u>stop talking to</u> that commotion, it seems to have been bothering him for a while. He looks around, and a loud clatter, then shakes his head, clearly confused. "Hey, no... <u>I've been out of town</u>, so I don't see much of this, but I think we could get there. You should see if anyone's coming over there, I don't know if I've seen anyone lately... but I think it's best if you go talk to that commotion and see if there's anyone out there. (Bad coherence, some conflicting parts are underlined)</p>
<p>➤ Persona Consistency: Consistency between the generated sentences and the given persona. Specifically, a persona consistent story should reflect the desired persona of the character implicitly by plots or his/her actions.</p>	
Options	1. System A is better 2. System B is better 3. System A and System B are good or bad equally.
e.g.	<p>Persona: [Anthony] The ghosts scare you. The moaning houses terrify you. The darkness... well, who knows what could be in it?</p> <p>Context: The InSpectres made their way into the woods, only the vaguest hint of a trail in front of them. The further they went in, the more the trail disappeared. Despite going in midday, a fog seemed to cover the forest. The fact that there was any light at all was the only sign that it was still day beyond the trees. Before them, broken branches and footprints seemed to be the only signs that the creature had gone in front of them. They had to follow the path to find its home. Hopefully it would be easy...But a job like this is never easy. Even as they stepped forward, they could see and hear movement beyond the fog... Something dangerous was out there. It would definitely hamper their search if it wasn't dealt with properly...</p> <p>Generated Story 1: ...There! Something moved, he was sure of it! "Were cat! Were cat! It's here!" "Actually, no," Ronnie observed once it hopped out of the fog... <u>"it'll change then kill us!"</u> .. (Good persona consistency, personalized sentence is underlined)</p> <p>Generated Story 2: His eyes narrowed as he could see through the mist, and his hands danced like a pair of marionettes in the moon. Slowly he started to climb, until he felt his hands begin to brush against the dark fog. It was like the sun was setting. He noticed a light shining outwards, and something...something was there. God he was glad he didn't fall into the same pit. (Bad persona consistency: reflect no persona)</p> <p>Generated Story 3: "A wolf?" Aiden yelled. <u>He felt the urge to make himself sound faster and faster at the end of his body.</u> "It's ok." "No." The wolf said. The other thing began shouting through the forest. "It's not like this. You're not a bad man." (Bad persona consistency: reflect calm persona, personalized sentence is underlined)</p>

Figure 3: The guideline of story quality evaluation.

Error Type	Cases
Opposite	<p>Context: The students have a surprisingly restful night of sleep. No weird dreams, no monsters, no strangers in tuxedos. The students head out to school. They run into each other as they enter the building and decide to have a chat about yesterdays events over breakfast...The students grab what they want to eat and sit at a table near the back of the cafeteria. They agree to try to keep calm in order to not raise the suspicion of other students and faculty.</p>
	<p>Persona: [Vivian] You are frequently lost in thought making you slow to action.</p>
	<p>Story: ...It's a big space, so that's a great opportunity to get excited, to get to know some of the new people. You may not be the most social person around, but you're here for fun..."</p>
	<p>Plan keywords: excit→student→like→come→...</p>
Irrelevant	<p>Context: The lights are bright above you, you lay in a bed with metal bars on either side. At first you think you are in a hospital but it did not smell like one, in fact you could smell nothing at all. When you lift your head you see that there are seven other people in the room with you and you are all wearing white clothing that looks a lot like scrubs but made of much better fabric...There is a doorway that leads into a hall, there is no door just an open doorway and the hall looks to be the same as the room.</p>
	<p>Persona: [Shay Lorence] You are not anxious and it is easy for you to believe someone else</p>
	<p>Story: <i>What's going on here? ... Looking up the hall, the room had black or gray light. The dark looked like that from the darkest rooms in the world, or maybe it could be dark, darker. I could see it from the window, the darkness in the room, even the bed, but that couldn't be the case. There was nothing here. There was just a room with no light. ... I had to take the other side to the hall, to the other side of the hall. The hallway was endless. That was the hardest thing to see...</i></p>
	<p>Plan keywords:room→bed→nightmare→wall→hallway→wake...</p>

Figure 4: Typical errors by our model. **Bold** words indicate the events exhibiting the opposite persona. *Italic* words indicate the events that are irrelevant with the given persona. And the **red** sentence indicate the generated target by CONPER.