Controllable Multi-Character Psychology-Oriented Story Generation

Feifei Xu
Shanghai University of Electric Power
xufeifei@shiep.edu.cn

Xinpeng Wang
Shanghai University of Electric Power
wangxinpeng@mail.shiep.edu.cn

Yunpu Ma*
Ludwig Maximilian University
cognitive.yunpu@gmail.com

Volker Tresp
Siemens Corporate Technology
Ludwig Maximilian University
volker.tresp@siemens.com

Yuyi Wang
ETH Zurich, Switzerland
yuwang@ethz.ch

Shanlin Zhou
Shanghai University of Electric Power
zhoushanlin@mail.shiep.edu.cn

Haizhou Du
Shanghai University of Electric Power
duhaizhou@shiep.edu.cn

ABSTRACT

Story generation, which aims to generate a long and coherent story automatically based on the title or an input sentence, is an important research area in the field of natural language generation. There is relatively little work on story generation with appointed emotions. Most existing works focus on using only one specific emotion to control the generation of a whole story and ignore the emotional changes in the characters in the course of the story. In our work, we aim to design an emotional line for each character that considers multiple emotions common in psychological theories, with the goal of generating stories with richer emotional changes in the characters. To the best of our knowledge, this work is first to focuses on characters' emotional lines in story generation. We present a novel model-based attention mechanism that we call SoCP (Storytelling of multi-Character Psychology). We show that the proposed model can generate stories considering the changes in the psychological state of different characters. To take into account the particularity of the model, in addition to commonly used evaluation indicators(BLEU, ROUGE, etc.), we introduce the accuracy rate of psychological state control as a novel evaluation metric. The new indicator reflects the effect of the model on the psychological state control of story characters. Experiments show that with SoCP, the generated stories follow the psychological state for each character according to both automatic and human evaluations.

CCS CONCEPTS

• **Computing methodologies → Natural language generation**.

*The corresponding author

KEYWORDS

Story Generation; Attention; Psychology; Character

1 INTRODUCTION

Table 1: An illustration of our story-generation system. The input sentence is the first sentence of the story, and our model appoints the characters like "Jervis" and "Girlfriend" and three psychological states of this story, like "Plutchik", "Maslow" and "Reiss". For Plutchik, every character has its own emotional lines and will change with our changing setup. The characters will share identical motivations based on Maslow and Reiss. Outputs of the model are the following four sentences that continue the story. The last box shows an example of the story.

| Input Sentence	Jervis has been single for a long time.	
Characters	Jervis, Girlfriend	
Plutchik	Jervis	sadness–anticipation–joy–joy–joy
	Girlfriend	none–none–joy–joy–joy
Maslow	love	
Reiss	contact, romance	
Story	**Jervis has been single for a long time.**	
	He wants to have a girlfriend.	
	One day he meets a nice girl at the grocery store.	
	They begin to date.	
	Jervis is happy that he is no longer single.	

"Storytelling is our speciality," the historian Yuval Noah Harari claimed in an interview, "It's the basis for everything we do as a species." And in his famous book, Sapiens: A brief history of humankind, how do human beings can control the planet? – Through

storytelling and gossip, by believing in shared fictions or myths. Even though one could disagree with his view, we can assume that a good storyteller can share her imagination and can use her emotions to infect others. The goal of this paper is to make machines learn to generate stories, a research topic in natural language generation. Story generation, in contrast to general text generation, requires coherence of texts in agreement with a sequence of events. Story generation is an important but challenging task in natural language generation, and it has also become one of the testing methods for advances in the field of AI. In addition to requiring coherence, there exist many challenges in story generation, such as thematic consistency, content consistency, word diversity, and sentiment control. There is considerable prior research on story generation, such as [4], [5] and [26]. However, these works focus only on how to generate a coherent, reasonable and diversified story. Recently, several works have developed controllable emotional text generation that can generate a story based on people's expected emotions, such as [8], [21] and [11]. Although these approaches can generate texts with different sentiments by appointing the sentiment to the story, they can consider only the sentiment of the whole story, and they are unable to control each character's sentiment in the course of the story.

Different from the above works, we propose a new model called SoCP (**S**torytelling **o**f Multi-**c**haracter **P**sychology), which generates a story by appointing the psychological states for each character, and assigns multiple emotions and motivations in accordance with specified psychological theories. An example of a generated story is shown in Table 1. It is a difficult task to consider and control multiple characters' psychological state lines simultaneously. In addition, psychological state lines normally involve multiple emotions based on the specified psychological theories. Compared to controlling a single character and the character's emotions in story generation, this is an extremely challenging task. To address these problems, we design two important components: the character selector and the psychological state controller. The character selector will select a current character and the corresponding psychological states from the psychological state line that was designed manually for multiple characters in the story and then proceed to the next component. The psychological state controller will accept the psychological states selected from the character selector, and then use an attention mechanism to control which psychological states and how many will be injected into the generating model. To account for generating a story that considers psychological changes in its characters, our designed model can generate a story by capturing each character's psychological state line even if it is sometimes implied and abstract.

The considered psychological theories are shown in Figure 1. We use two popular theories to describe the motivation of a person in Figure 1(a): the "hierarchy of needs" of Maslow [12] (left) and the "basic motives" of Reiss [18] (right). Maslow's "hierarchy of needs" has five categories from physiological needs to spiritual growth, providing a coarse-level representation. Reiss has 19 fine-grained categories that provide a more informative range of motivations, which can be considered to be a subset of the Maslow's motivations. The theories of emotion use the "wheel of emotions" of Plutchik [15] in Figure 1(b). It has eight basic dimensions of emotions to adequately portray a person.

Our contributions are as follows:
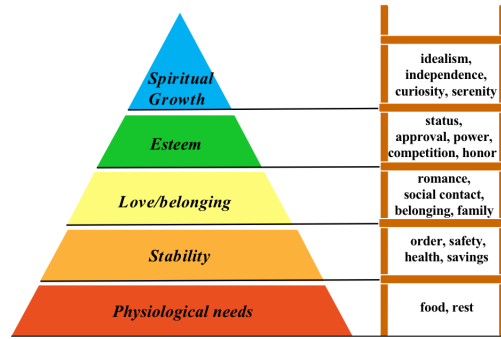
(a) Maslow (left) and Reiss (right)

(b) Pluchik

Figure 1: The psychological states of Maslow, Reiss and Pluchik. The pyramid in (a) shows the classification of Maslow from the physiological to the spiritual level. The right ladder represents the Reiss motives as a subset of Maslow's hierarchy of needs. Plutchik's wheel of emotions in (b) represents the eight basic dimensions of emotions.

- We automatically construct the Plutchik matrix, Maslow matrix, and Reiss matrix, which use a matrix mapping method to represent a person's psychological state.
- We propose a model called SoCP that will generate a story with multiple changes in emotion in characters based on the given characters and the corresponding psychological state lines. To ensure the coherence of the generated story and to achieve multi-character and multi-emotion control, we chose the Seq2Seq framework to generate the story. In addition, we design a character selector that determines which character should be selected to be described, and a psychological state controller based on an attention mechanism [1] that can control which and how many psychological states the model accepts.
- For evaluation, we develop an accuracy rate of psychological state control as a novel evaluation metric by using a psychology classifier. The experimental results show good performance in various automatic evaluating indicators (BLEU, ROUGE, etc.). The examples show that our model can generate fluent and reasonable stories based on the dataset.
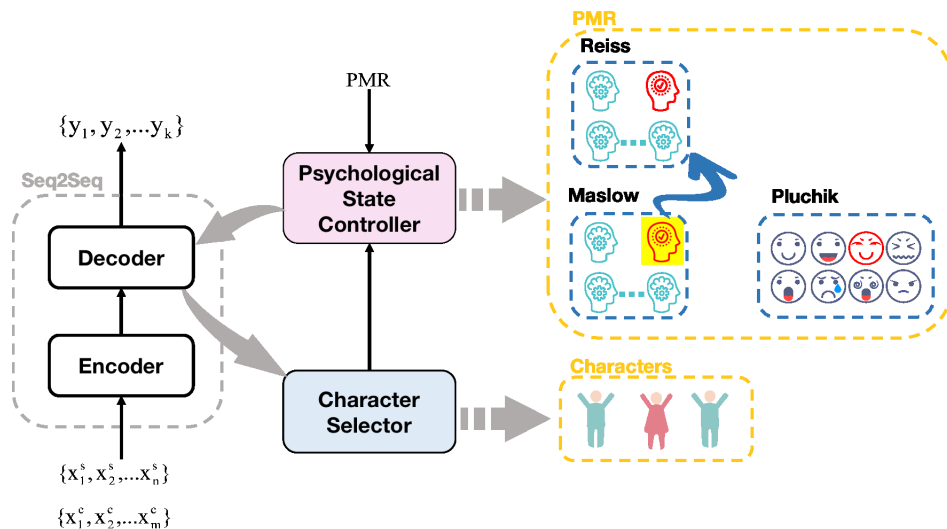
Figure 2: An Overview of the SoCP model, which based on the Seq2Seq structure. The PMR, or psychological state, is the input of the model. Two modules, the Character Selector and the Psychological State Controller, are designed to facilitate the generation of storylines with expected psychological states for each character in the story. The patterns in the yellow dotted line are illustrative descriptions of the two key modules in the model.

2 RELATED WORK

2.1 Story Generation

Story generation has attracted much attention over the past years. Recently, many novel ideas have been proposed. For instance, [9] generate a coherent story from independent descriptions, describing a scene or an event. [5] explore a strategy for story generation. Their frameworks both use sequence-to-sequence neural networks. Previous works have been experimentally proven that sequence-to-sequence neural networks [23] have good performance in text generation, e.g., in machine translation [1], summarization [19], and dialogue generation [21]. Considering the outstanding performance of the Seq2Seq structure, our model is designed-based on the sequence-to-sequence model. Our model is constructed with two specific components: the character selector and the psychological state controller. In addition, multiple emotional inputs based on psychological theory, such as the Plutchik matrix, the Maslow matrix, and the Reiss matrix are considered.

There are also many works using other structures. [4] use a hierarchical model that first generates a prompt, and then condition on the prompt when generating a story. [27] propose a multi-pass hierarchical conditional variational autoencoder model to enhance wording diversity and content consistency of the generated story.

2.2 Text Generation with Emotion

To generate a story considering our specific psychological theories, we introduce psychological states in addition to the actual sentence. There are also many works in text generation with emotion. [25] incorporate topic information into a sequence-to-sequence framework to generate informative and interesting responses for chatbots. [6] introduce a novel language model for generating affective conversational text conditioned on context words, an affective category

and an affective strength parameter. The model can generate expressive text at varying degrees of emotional strength without affecting grammatical correctness. Several works have been proposed in the area of emotional or sentimental dialogue generation [8, 28, 29]. They express emotion or sentiment by defining a binary sentiment label (positive/negative) or a set of emotions, such as "anger" and "love".

Different from these papers, we do not just add emotion to the story but assign different emotion lines to different characters. In particular, these emotions refer to the specified psychological theories, which include three indicators, and each indicator has many emotions or motivations that can describe a character more precisely. Thus, according to the particularity of this task, we choose a dataset from [17], which has many fine-grained psychologies for every character in the story. [14] used our dataset to extract, rank, filter, and classify sentiments in terms of their underlying human needs.

3 MODEL

3.1 Overview

An overview of our SoCP model is presented in Figure 2. Our model aims to generate a multi-character and multi-emotion story. In this task, we design story characters and the psychological state lines of each character as our input in addition to the seed text. We use a Seq2Seq structure to perform story generation based on the seed text, typically a single input sentence. For the purpose of controlling multiple characters and multiple emotions, we design two extra important components called the character selector and the psychological state controller. We first use the character selector to select a person and then use the psychological state controller to attend the three indicators: the Plutchik matrix, Maslow matrix
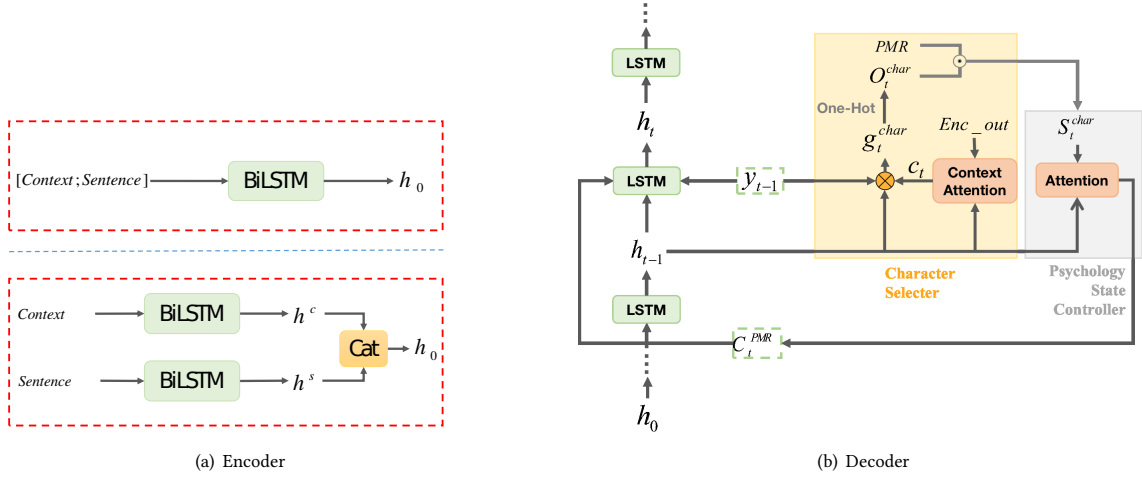
(a) Encoder

(b) Decoder

Figure 3: The encoder and decoder modules in the SoCP model. In the Encoder part, we present two methods to encode the text: Context-independent and Context-merge. In the Decoder part, we adopt Character Selector and Psychological State Controller to control text generation.

and Reiss matrix. For convenience, in this task, we simply refer to the psychological state as the emotion.

Formally, there is the input story sentence $X = (x_1^s, ..., x_n^s)$ consists of n words. To make the story coherent, we use the context $C = (x_1^c, ..., x_m^c)$ into account, which represents previous sentences of the input sentence X and consists of m words. We concatenate C and X as input. Meanwhile, there is each character's PMR score matrix: $S^{pmr} = (S_1^p, S_1^m, S_1^r), ..., (S_j^p, S_j^m, S_j^r)$, which represents the three Plutchik, Maslow, and Reiss indicators to describe a character in the story, where j is the number of characters in the story. Our purpose is to generate a sentence $Y = (y_1, ..., y_k)$ that describes the continuation of the story line. Essentially, our goal is to estimate the conditional probability $P(Y|[X, C], PMR)$, where PMR is the PMR matrix we designed. In this work, we assume that S^{pmr} are given as input for text generation.

3.2 PMR matrix

To add the psychological theories to story generation, we construct a PMR matrix, which is a concatenation of the Plutchik, Maslow, and Reiss matrices. The three indicators represent the psychological state of a character in a storyline. In the dataset, each indicator has its own score annotated by human annotators, and we use these scores to construct a vector for each indicator. Due to the different number of characters in different stories, we fix the number of characters as the maximal number of characters in all stories. A multiple characters score matrix is obtained by concatenating the Plutchik score S^p, Maslow score S^m, and Reiss score S^r. Their dimension is the maximum number of characters multiplied by the number of psychological states. Then, we randomly initialize a word embedding matrix for three indicators as V^p, V^m, and V^r respectively, and their dimension is the number of psychological states multiplied by the size of the word embedding. We multiply the word embedding matrix with multiple characters score matrix and then map them onto a feature space. Subsequently, we obtain

the Plutchick, Maslow, and Reiss matrices as follows:

$$P_i = W_p(S_i^p \times V_i^p) + b_p, \tag{1}$$

$$M_i = W_m(S_i^m \times V_i^m) + b_m, \tag{2}$$

$$R_i = W_r(S_i^r \times V_i^r) + b_r, \tag{3}$$

where W_p, W_m, W_r are weight matrices for deriving Plutchik, Maslow, and Reiss representations of characters, respectively. b_p, b_m, b_r are biases and i indicates the i-th character. For the convenience of calculation, we can concatenate the Plutchik, Maslow, and Reiss matrices as multiple characters PMR matrix:

$$PMR_i = W_{pmr}(S_i^{pmr} \times V_i^{pmr}) + b_{pmr}, \tag{4}$$

where W_{pmr} and b_{pmr} are learnable weight matrix and bias.

3.3 Encoder

3.4 Encoder

For the coherence of the generated story, we present two methods to add contextual information into the model. We use a normal bidirectional **LSTM** [20] as our encoder module, which is illustrated in Figure 3(a). We do not only encode the input sentence but also consider the contexts, which include the previously generated sentences.

3.4.1 Context-independent. The first method, called the context-merger method, concatenate the context and sentence together as input which form the input to the first BiLSTM encoder. Subsequently, we obtain the last hidden state of encoder h_0.

$$x = (x_0^c, x_1^c, ..., x_m^c; x_0^s, x_1^s, ..., x_n^s), \tag{5}$$

$$h_0 = \text{BiLSTM}(x). \tag{6}$$

3.4.2 Context-merge. In addition, we use two BiLSTM encoders to encode the sentence and context. Then, we concatenate the last hidden state h^c of the encoded context and h^s of the encoded sentence as text representation that is input to the decoder, which is also the initial hidden state h_0 of the decoder.

$$h^c = \text{BiLSTM}^c(x_0^c, x_1^c, ..., x_m^c), \tag{7}$$

$$h^s = \text{BiLSTM}^s(x_0^s, x_1^s, ..., x_n^s), \tag{8}$$

$$h_0 = [h^c; h^s], \tag{9}$$

where the $[\cdot; \cdot]$ represents the concatenation operation.

3.5 Decoder

The structure of our decoder is shown in Figure 3(b). In the decoder part, we have two auxiliary modules to help generate sentences: the character selector and the psychological state controller.

3.5.1 Character Selector. We design a character selector to select which person should be described during the decoding. At each step t, we apply Multi-Layer Perceptron (MLP) to compute a character gate g_t^{char} with the input of the word embedded in the previously decoded word y_{t-1}, the previous hidden state of the decoder h_{t-1}, and the current context vector c_t. The character gate is defined as follows:

$$g_t^{char} = \text{softmax}(W_g[y_{t-1}; h_{t-1}; c_t]), \tag{10}$$

$$o_t^{char} = \text{onehot}(g_t^{char}), \tag{11}$$

$$s_t^{char} = o_t^{char} \times PMR_i, \tag{12}$$

where W_g is the weight matrix in character gate. We use softmax activation to select a character of maximum probability. Then, we transform it into a one-hot vector. Finally, PMR_i is multiplied by the one-hot vector to obtain the final psychological state of the selected character.

The above character gate mechanism indicates that the selected character that would be described in step t in the decoding.

3.5.2 Psychological State Controller. After the character is selected by the character selector, we use a psychological state controller inspired by [1] to control which psychological states and how many we accept. The formulation of PMR context c_t^{PMR} at step t is as follows:

$$e_{t,i} = V_a^T \tanh(W_a s_t^{char} + U_a h_{t-1}), \tag{13}$$

$$\alpha_{t,i} = \frac{\exp(e_{t,i})}{\sum_{i=1}^{c} \exp(e_{t,i})}, \tag{14}$$

$$c_t^{PMR} = \sum_{i=1}^{c} \alpha_{t,i} s_t^{char}, \tag{15}$$

where V_a^T, W_a, U_a are learnable parameters, and c is the number of characters. Through our psychological state controller, the model is effective in learning which psychological state should be attended.

3.5.3 State Updating. We use LSTM [7] with attention as the decoder. We initialize the hidden state of the decoder with the encoded text representation h_0 from the encoder output. The decoder input is the concatenation of the last timestep word embedding vector y_{t-1} and PMR context c_t^{PMR}, which represent adding the psychological state into each character.

We update the LSTM hidden state as follows:

$$h_t = \text{LSTM}([y_{t-1}; c_t^{PMR}; c_t], h_{t-1}), h_{t=0} = h_0. \tag{16}$$

3.6 Training

Our training objective is the negative log likelihood.

$$L_{NLL} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} logP(y_t^{(i)}|y_{<t}^{(i)}, X^{(i)}, C^{(i)}, P^{(i)}, M^{(i)}, R^{(i)}),$$
$$\tag{17}$$

where N is the total number of the dataset and T is the timestep of the i-th output sentence. $X^{(i)}$ represents the i-th sentence in the dataset. Similarly, $C^{(i)}$, $P^{(i)}$, $M^{(i)}$ and $R^{(i)}$ represents the i-th context, i-th Plutchik matrix, i-th Maslow matrix and i-th Reiss matrix in the dataset respectively.

We minimum the objective function with the given inputs to generate the proper sentences.

4 EXPERIMENTS

4.1 Dataset

We choose a story corpus that has been annotated with psychology theories [17] , and consists of 4k five-sentence stories. Therefore we have 20k sentences. Each sentence is annotated with characters and three psychological theory indicators. In psychological theory, Plutchik has 8 fine-grained categories, Maslow has five coarse-grained categories, and Reiss has 19 fine-grained categories. Figure 4 shows the statistics of the psychological states. It shows that Plutchik is the most frequently annotated psychological state compared to Maslow and Reiss. In particular, among the annotated Plutchik states, "joy" and "participation" are the most frequent states. Specifically, the Reiss categories can be considered as subcategories of the Maslow categories. We have processed the dataset by different methods for the *Plutchik*, *Maslow* and *Reiss* indicators in the following way. Since the original data are annotated by three workers and since different workers have their own viewpoint, we add them together and obtain only the score that is marked by more than two workers for the Plutchik indicator and then we normalize it. The Maslow and Reiss are represented as one-hot vector. We randomly split the data, and use 80% of the data as training set and 20% as test set. However, due to the lack of enough stories, we adopt a classifier proposed in [14] to label the psychological states of characters in the unannotated stories and use them as augmented data. By including the augmented data, we have approximately 15k stories in total. In the test phase, we use the beginning sentences, contexts, and the normalized scores of psychological states as inputs. The character number statistics are shown in Table 2. The upper limit of the number of characters is six, and the number of characters in a story is mostly distributed in the range of 1-3. Thus,
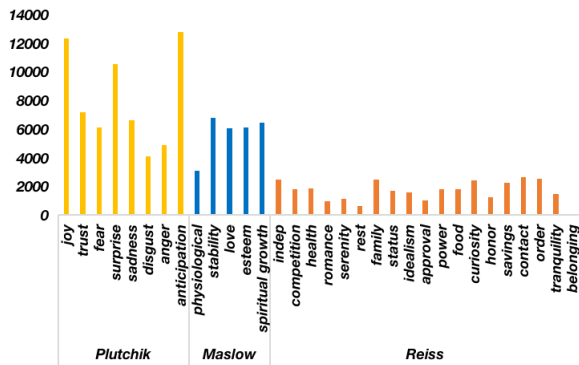
Figure 4: The statistic of psychological states. "Anticipation" is the most frequent state in Plutchick's wheel of emotion, while "belonging" is the least frequent state in the Reiss classification.

Table 2: The statistic of the number of characters in sentences.

Character number	Sentence number
1	11008
2	6547
3	1254
4	164
5	36
6	1

we fix the number of characters to 3, which covers most of the training data.

4.2 Baselines

To evaluate the effectiveness of the SoCP framework, we compare our methods against representative baselines as follows:

Seq2Seq is a generation model that has good performance in the translation task and other generation tasks. It is also the foundation model of our SoCP. Therefore, we use the Seq2Seq baseline as a comparison metric to prove whether our model can improve the traditional Seq2Seq and whether it exerts a generation effect on fluency.

Inc-S2S, mentioned in [26], denotes the incremental sentence-to-sentence generation baseline. Our experiment differs from this implementation in that we do not have the story title, we appoint the psychological states to the model, and we generate the next sentence from the given first sentence of the story and previously generated sentences. We also use the Seq2Seq model, paying attention to construct the Inc-S2S baseline. In this way, we can prove whether our character selector and psychological state controller work effectively.

Transformer architecture [24] has attracted the attention of many scholars in recent years. It greatly improves the state of the art in natural language processing tasks by training on a huge amount of data with large-scale language models (LMs), such as BERT [3], GPT-2 [16], and Transformer-xl [2].

GPT-2 [16] is well known in natural language processing models in recent years because of its huge number of parameters, and it also composed of a Transformer structure. Many works have proven that GPT-2 can generate a reasonable and personalized story. Therefore, we use GPT-2, a representative of story generation models, as a benchmark to our model.

4.3 Experimental Settings

Following the discussion above, we fix the number of characters to 3 and use 'none' to represent the character if the number of characters is smaller than 3. 300-dimensional pre-trained glove vectors are used as the word embeddings. The encoder is designed as a 2-layer bidirectional LSTM with a hidden size of 256 and the decoder as a 1-layer LSTM with a hidden size of 256. The PMR matrix is used to from a high-dimensional to a low-dimensional vector with dimension 256. The batch size is 8, and the dropout [22] is set as 0.2. We use the Adam optimizer [?] with an initial learning rate of 0.0003.

4.4 Evaluation Metrics

BLEU: For each story in the testset, we take the input sentence x and the human-annotated processed PMR score S^{pmr} as input. The corresponding label is $\hat{y}$. Then, we calculate the BLEU [13] score of the generated y and $\hat{y}$ as the overall quality of the generated story.

ROUGE: Similar to BLEU, ROUGE is also a common evaluation indicator in machine translation and text summarization. It is used to measure the similarity between automatically generated and reference results.

Meteor: Meteor considers word shape and is also an important evaluation indicator of fluency.

ACER (Accuracy of Controlling Emotions of Roles): We use BiLSTM to pretrain a classifier to classify the psychological state of each character in the generated sentences. It indicates our model's capability of expressing emotion. Because of the complexity of psychological state, we design a classifier only for the Plutchik indicator. We concatenate the name of a story character $Char$ and the sentence X as the input. In this way, many training pairs in identical sentences with different outputs can be obtained because different characters have different psychological states in a sentence. Through BiLSTM, we can obtain the compressed vector h^{clf}, and then, we use two fully connected layers to map it onto the output size.

$$h^{clf} = \text{BiLSTM}([Char; X]), \tag{18}$$

$$p = W_p(\text{ReLU}(W_o(h^{clf}) + b_o)) + b_p, \tag{19}$$

where W_p, W_o, b_o, and b_p are learnable parameters.

4.5 Evaluation

To demonstrate the effectiveness of our components, we compare our model with the Seq2Seq structure, the Inc-S2S and Transformer baselines in the BLEU, ROUGE, and Meteor automatic evaluation metrics and our designed metric ACER, which indicates the accuracy with which the psychological state of the generated sentence is consistent with our previous set. As shown in Table 3, every criteria score in our model is the maximum, which indicates the

Table 3: Automatic evaluations of the proposed model and the baseline models. The last line with * indicates that the results are improved using data augmentation.

Model	BLEU				ROUGE			Meteor	ACER
	B-1	B-2	B-3	B-4	R-1	R-2	R-l		
Seq2Seq	0.202	0.049	0.016	0.007	0.122	0.014	0.117	0.062	0.654
Seq2Seq+context-merger	0.208	0.049	0.016	0.008	0.134	0.012	0.126	0.065	0.674
Seq2Seq+context-independent	0.223	0.055	0.018	0.009	0.147	0.016	0.138	0.068	0.684
Transformer+context	0.207	0.053	0.020	0.008	0.139	0.012	0.128	0.069	0.747
Inc-S2S+context	0.224	0.053	0.017	0.006	0.151	0.013	0.141	0.067	0.825
(Ours)SoCP+context-merger	0.216	0.056	0.021	0.010	0.144	0.016	0.136	0.067	0.886
(Ours)SoCP+context-independent	0.232	0.062	0.025	0.011	0.161	0.018	0.151	0.072	0.879
(Ours)SoCP+context-independent*	**0.241**	**0.070**	**0.029**	**0.014**	**0.167**	**0.022**	**0.158**	**0.076**	**0.893**

(a) attention map 1

(b) attention map 2

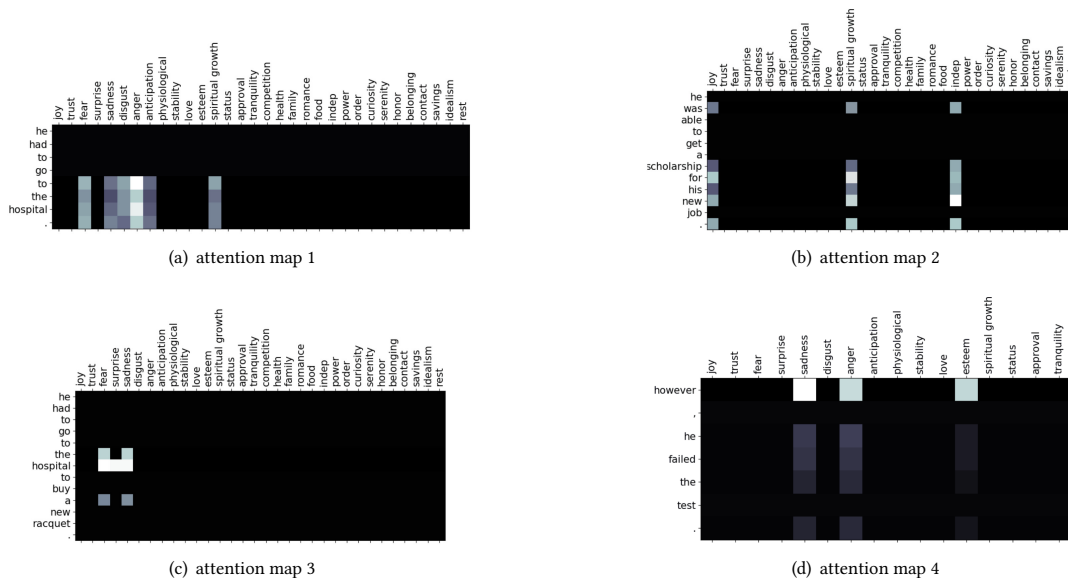(c) attention map 3

(d) attention map 4

Figure 5: Visualization of our psychological state controller. The row in the figure represents the fine-grained emotion and motivation of the three psychological states, and the column is the generated sentence. The brighter the grid in the graph is, the more there is attention between rows and columns.

effectiveness of our designed modules, and the generated sentences are smooth.

For BLEU, ROUGE and Meteor, we see that Seq2Seq has better performance than the Transformer structure. The reason may be that most of the sentences in the data are short text, and it is more suitable to use Seq2Seq. Additionally, it is better to use the context-independent method than the context-merge method with Seq2Seq, which is mentioned in the encoder part. Of all the models, our proposed model has the best performance.

For ACER, which we design to test whether the emotion of the sentences we generated is consistent with our previous settings. Inc-S2S is better than other models but worse than our model, which proves that our proposed model with our designed component works effectively.

4.6 Interpretability

In this part, we study the learned attention distributions of characters' psychological states between generated sentences and three psychological theories, i.e., of Plutchik, Maslow and Reiss. The model provides interpretability in two ways: by selecting a character based on the character selector using Eq.10 and by choosing the psychological state based on the psychological state controller using Eq.14, as shown in Figure 5. The lighter the square at the intersection between two words, the more attention the model pays to psychological states when generating the next word.

4.6.1 The Effect of the Character Selector.
As previously mentioned, we proposed the character selector module to select a character that we would like to describe in the current timestep. Visualization

Table 4: Examples generated by the proposed model and the baseline models.

	Jane bought a new necklace.	The man next door seemed to be losing his mind.
Input	Characters: Jane, Friend Plutchik: joy-joy-joy-joy-joy, none-none-joy-none-none Maslow: esteem, love Reiss:approval	Characters: Man Plutchik: fear-disgust-surprise-surprise-joy Maslow: spiritual growth Reiss: status
Seq2Seq	Jane bought a new necklace. She was running it. she was amazed it. She tried it. She was thankful jane it.	The man next door seemed to be losing his mind. He was very happy to have a new job. He was not sure what he was going. He was not sure what he was going. He was not sure to go.
Inc-S2S	Jane bought a new necklace. She was not sure what she wanted to go to the store. She was very happy. She was very happy. She was so happy that she had a good time.	The man next door seemed to be losing his mind. He asked his credit hour he did not know how. He asked his parents would help him some directions. He tied the games and tied miles. He was sad he realized he realized that his card.
Transformer	Jane bought a new necklace. She decided to do something about it. She did not have any money. She decided to save money. She decided to save up.	The man next door seemed to be losing his mind. He yelled at him. He had to go to the hospital. He asked his girlfriend. She said yes right away.
GPT-2	Jane bought a new necklace. "I'm not sure if I'm going to be able to wear it anymore," she said. "I'm not sure if I'm going to be able to wear it anymore." She said she's been looking for a way to get rid of the necklace. "I'm not sure if I'm going to be able to wear it anymore," she said.	The man next door seemed to be losing his mind. "I'm sorry, but I'm not going to be able to go to the hospital," he said. "I'm not going to be able to go to the hospital." The man next door was also in shock. "I'm not going to be able to go to the hospital," he said.
SoCP	Jane bought a new necklace. She was excited to get a job. Her friend was on the same. She was very happy. She was happy to the best game.	The man next door seemed to be losing his mind. He was very angry and did not want to do. He was a detective man about the problem. The man was not as the man was not the wrong. He was able to beat the ball and he was shot.

Table 5: Examples of controllable generated stories

Input sentence	Jane bought a new necklace.	
Character	Jane	
Maslow	esteem, love	
Reiss	approval	

Generated stories with the same Plutchik indicator "joy" under different scores		
joy=1	**joy=0.5**	**joy=0**
Jane bought a new necklace. She was a good amused. She went to school. She was a happy. She was proud of her career.	**Jane bought a new necklace.** She was a really excited. She decided to have a swim. She did not think it was on it. It was the best she ever.	**Jane bought a new necklace.** She needed to go to a different game for a month. She was nervous about her feet. She was so tired, they got there for the store. Her mom was not happy.

Generated Stories with Different Plutchik Indicators		
surprise=1	**fear=1**	**anger=1**
Jane bought a new necklace. She was surprised to see what she had been. The day was a huge. I did not know how to stop. I was shocked when i looked at the door.	**Jane bought a new necklace.** She had no idea of her own. She was afraid of heights. She had to swim on stage. She had to be scared to be scared.	**Jane bought a new necklace.** She was angry at a slow game. The kid removed the flame on the drums. it was too late. It turns out a knife to be an important.

Generated Stories with Multiple Plutchik Indicators		
trust=1, joy=0.5	**sadness=0.4, anticipation=0.8**	**fear=1, disgust=0.5**
Jane bought a new necklace. She was a book. She asked her mom to borrow some advice. Her boyfriend bought her a chinese outfit. Her boyfriend promised her to teach her hair.	**Jane bought a new necklace.** She had never been very busy. Her husband had crashed by. Her mother had crashed. her mother had to be an prepared.	**Jane bought a new necklace.** She had a tooth coming in a dark. She was nervous, but she was going to be late. She was afraid of heights. She was afraid of the back, but she felt very scared.

Table 6: Examples of controllable generated stories

Input sentence	Tiffany was a defensive soccer player who had never scored a goal.		
Character	Tiffany		
Maslow	esteem		
Reiss	competition		

Generated stories with the same Plutchik indicator "joy" under different scores		
joy=1	**joy=0.5**	**joy=0**
Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...
She was so excited and decided to go see the road.	She was a total with the sidewalk.	She was not a personal when he heard a running to the sink.
The resulting were very little and she visited.	She was a very hard.	She was a bit scared, but the water was not working.
It was a great success.	She was able to get things from work.	The manager exchanged the product , but was hard.
I was glad that she had learned to pay a new shoes.	She was glad he had not clean it!	The manager had to move, and they were able to meet.
Generated Stories with Different Plutchik Indicators		
surprise=1	**fear=1**	**anger=1**
Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...
She was surprised to see what he was going to go.	She had been afraid of them since there was no signal.	She tried to use the bigger, but the computer got very hot.
She was halfway there, and the power was coming up.	She tried to be careful when she was there to be at last.	The computer broke down and the bread broke down.
The soup was a bit, but it almost!	she was so nervous, because she had to be stuck in her bed.	It was also a long time for the driver.
She was shocked to see her daughter.	she was very scared, but the distance went through the door.	It was the last time ever ever could not play football.
Generated Stories with Multiple Plutchik Indicators		
trust=1, joy=0.5	**sadness=0.4, anticipation=0.8**	**fear=1, disgust=0.5**
Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...	Tiffany was a defensive soccer player who ...
She was a simple with the paper.	She had not been very busy at all, and watched a year.	She had been afraid of food since there was no news.
She decided to make a budget.	She asked for a long time to play for 10 minutes.	She tried to be careful when she was all over town.
She was very well.	The hiring on the next day, She was nervous for almost.	She was afraid that she would be an hour for there.
She was able to borrow a new tiesa.	The irony was a janitor that was looking at the end.	She was very scared and scared of the safety!

of the attention maps provides evidence of the ability of the model to capture relevant psychological states that belong to each person. Each word in the sentence corresponds to squares of different colors, indicating that our component can automatically read the psychological state of multiple characters we input and can automatically choose which character's psychological state to accept. The black square indicates that the character selector does not obtain any responsible psychological state because not all words, such as 'a' and 'the', necessarily express the emotion. In summary, our model can be controlled by our designed component to accept external information, similar to our psychological state.

4.6.2 The Effect of the Psychological State Controller. Figure 5 shows examples of which psychological state and how much can be attended by the psychological state controller. In the examples shown, the model correctly picks the indicators from the input psychological state. As shown in Figure 5, the first attention map has the focus on the Plutchik's emotions, such as "fear" and "anger", while the second attention map attends to the indicators of Maslow and Reiss, such as "spiritual growth" and "indep". In the third attention map, the word "hospital" is associated with the Plutchik indicators such as "fear", "surprise", and "sadness", which indicates that "hospital" is normally related to the negative emotions of a character. The word "however" expresses a strong turning point, and it said that he failed the test. The word "however" foretells a strong turning point and sad results that the character will fail the test in the hospital, which is also consistent with the "sadness" and "anger" emotions that we input.

4.7 Case Study

4.7.1 Comparison with Baseline Models. Table 4 shows the examples of stories generated by our model and the baseline models. Our model can generate a story that is coherent and consistent story with the psychological state we previously provided.

For the Seq2Seq model, we see that it usually generates repeated sentences, and it does not consider all the characters. Example 1 in Table 4, generated by the baseline models describe only one character, "Jane", but, for the SoCP can generate information about the "Friends". In example 2, we see that the baseline models cannot vary the story with our given psychological state and even has wrong emotions, but our SoCP model can fit it properly. The GPT-2 model can generate reasonable sentences but has many repetitions.

Overall, our proposed model can generate stories by controlling characters' emotions. However, we see that not all stories are coherent; thus, the coherence is also a challenge confronting our model.

4.7.2 Controllability Over Stories. In Table 5 and Table 6, we show examples of controllability generated stories under different conditions. The first example in Table 5 compares the generated stories using an identical Plutchik indicator however under different scores. In particular, in the first example, we assign the Plutchik indicator "joy" with different scores: 1, 0.5 and 0. When the score is equal to 1, it generates some obvious words, such as "good", "happy", and "was proud of". As the score of the 'joy' is lower, the generated words are more and more negative. When the score is equal to 0, it generates some negative words, such as "nervous", "tired", and "not happy". The second example shows generated stories with different Plutchik indicators. We assign "surprise", "fear", and "anger" as different Plutchik indicators. When Plutchik is "surprise", it generates some words, such as "was surprised to", and "shocked". When Plutchik indicates "fear", it generates some words, such as "was afraid of", and "be scared to". When Plutchik indicates "anger", it generates some words, such as "angry". For the third example, we assign multiple Plutchik indicators of different scores. We can see the generated stories have multiple emotions.

In Table 6 , we show the different examples of controllable generated stories. Similarly, they have an obvious effect. According

to the examples shown above, we see the controllability of our model. There exist some stories that are not coherent in the above examples, which indicate our model pays more attention to the psychological states and overlooks the coherence of the story. This is where we need improvement.

5 CONCLUSION

Traditional story generation models show poor performance because they only take into account emotional changes in story development without considering the internal emotional and motivational changes of characters. However, a story is usually written according to people's ideas and settings. Therefore, we have proposed a method that can specify the psychological changes in different characters in a story. In the application of the model, the psychological state can be assigned to the psychological robot so that it can generate helpful words for people with mental diseases. For future work, we will add an external knowledge graph to our model to better control word generation, and we will use an advanced transformer model to represent the text.

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate, 2014.

[2] Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc Le, and Ruslan Salakhutdinov. Transformer-XL: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988, Florence, Italy, July 2019. Association for Computational Linguistics.

[3] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[4] Angela Fan, Mike Lewis, and Yann Dauphin. 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, July 2018. Association for Computational Linguistics.

[5] Angela Fan, Mike Lewis, and Yann Dauphin. Strategies for structuring story generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2650–2660, Florence, Italy, July 2019. Association for Computational Linguistics.

[6] Sayan Ghosh, Mathieu Chollet, Eugene Laksana, Louis-Philippe Morency, and Stefan Scherer. Affect-LM: A neural language model for customizable affective text generation. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 634–642, Vancouver, Canada, July 2017. Association for Computational Linguistics.

[7] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[8] Chenyang Huang, Osmar Zaïane, Amine Trabelsi, and Nouha Dziri. Automatic dialogue generation with expressed emotions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 49–54, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[9] Parag Jain, Priyanka Agrawal, Abhijit Mishra, Mohak Sukhwani, Anirban Laha, and Karthik Sankaranarayanan. Story generation from sequence of independent short descriptions. *ArXiv*, abs/1707.05501, 2017.

[10] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization, 2014. cite arxiv:1412.6980Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.

[11] Fuli Luo, Damai Dai, Pengcheng Yang, Tianyu Liu, Baobao Chang, Zhifang Sui, and Xu Sun. 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, July 2019. Association for Computational Linguistics.

[12] Abraham Harold Maslow. A theory of human motivation. *Psychological review*, 50(4):370, 1943.

[13] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 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, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics.

[14] Debjit Paul and Anette Frank. Ranking and selecting multi-hop knowledge paths to better predict human needs. 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 3671–3681, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics.

[15] Robert Plutchik. A general psychoevolutionary theory of emotion. *Theories of emotion*, 1:3–31, 1980.

[16] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8):9, 2019.

[17] Hannah Rashkin, Antoine Bosselut, Maarten Sap, Kevin Knight, and Yejin Choi. Modeling naive psychology of characters in simple commonsense stories. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2289–2299, Melbourne, Australia, July 2018. Association for Computational Linguistics.

[18] Steven Reiss. Multifaceted nature of intrinsic motivation: The theory of 16 basic desires. *Review of general psychology*, 8(3):179–193, 2004.

[19] Alexander M. Rush, Sumit Chopra, and Jason Weston. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal, September 2015. Association for Computational Linguistics.

[20] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.

[21] Zhenqiao Song, Xiaoqing Zheng, Lu Liu, Mu Xu, and Xuanjing Huang. Generating responses with a specific emotion in dialog. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3685–3695, Florence, Italy, July 2019. Association for Computational Linguistics.

[22] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[23] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 2*, NIPS'14, page 3104–3112, Cambridge, MA, USA, 2014. MIT Press.

[24] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, undefinedukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc.

[25] Chen Xing, Wei Wu, Yuehua Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. Topic augmented neural response generation with a joint attention mechanism. *ArXiv*, abs/1606.08340, 2016.

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

[27] Meng-Hsuan Yu, Juntao Li, Danyang Liu, Bo Tang, Haisong Zhang, Dongyan Zhao, and Rui Yan. Draft and edit: Automatic storytelling through multi-pass hierarchical conditional variational autoencoder. *AAAI*, 2020.

[28] Hao Zhou, Minlie Huang, Tianyang Zhang, Xiaoyan Zhu, and Bing Liu. Emotional chatting machine: Emotional conversation generation with internal and external memory. *AAAI*, 2018.

[29] Xianda Zhou and William Yang Wang. MojiTalk: Generating emotional responses at scale. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1128–1137, Melbourne, Australia, July 2018. Association for Computational Linguistics.