Research Article

Story Generation Using Knowledge Graph under Psychological States

Feifei Xu, Xinpeng Wang, and Shanlin Zhou

College of Computer Science and Technology, Shanghai University of Electric Power, Shanghai, China

Correspondence should be addressed to Xinpeng Wang; wangxinpeng@mail.shiep.edu.cn

Received 18 February 2021; Revised 17 March 2021; Accepted 29 March 2021; Published 28 April 2021

Academic Editor: Honghao Gao

Copyright © 2021 Feifei Xu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Story generation, aiming to generate a story that people could understand easily, captures increasing researchers’ attention in recent years. However, a good story usually requires interesting and emotional plots. Previous works only consider a specific or binary emotion like positive or negative. In our work, we propose a Knowledge-Aware Generation framework under Controllable Conditions (K-GuCCI). The model assigns a change line of psychological states to story characters, which makes the story develop following the setting. Besides, we incorporate the knowledge graph into the model to facilitate the coherence of the story. Moreover, we investigate a metric AGPS to evaluate the accuracy of generated stories’ psychological states. Experiments exhibit that the proposed model improves over standard benchmarks, while also generating stories reliable and valid.

1. Introduction

Story generation has been an emerging theme in natural language processing technologies [1–3]. Much of the research has examined the coherence, rationality, and diversity of the generated stories. Huang et al. and Song et al. [4, 5] and Luo et al. [6] argued that assigning emotions in text generation could enrich the texts and full of variety. From psychological theories, Figure 1 shows the fine-grained psychological states involved in an individual. Figure 1(a) displays the motivation of folks described by the two popular theories: the “Hierarchy of Needs” of Maslow [7] on the left, and the “Basic Motives” of Reiss [8] on the right. “Hierarchy of Needs” proposed by Maslow use such terms as physiological needs, stability, love/belonging, esteem, and spiritual growth to describe the evolution of human motivation. There are nineteen fine-grained categories that include a variety of motives in Reiss, which is richer than that in Maslow. Plutchik [9], also called the “Wheel of Emotions”, has eight motions shown in Figure 1(b). Xu et al. [10] propose a novel model called SoCP, which uses the theories to generate an emotional story. Nevertheless, it still lacks richness and coherence.

Knowledge graph, known for better semantic understanding, is a key factor in the success of natural language processing. External knowledge can be introduced to increase the richness of the texts in story generation. Zhou et al. [11] use a large-scale commonsense knowledge in neural conversation generation with a Graph Attention approach, which can better interpret the semantics of an entity from its neighboring entities and relations.

Accordingly, we propose a model called K-GuCCI, which leverages the knowledge graph to enhance the coherence of story generation and psychological theories to enrich the emotion of stories. Table 1 shows an example of a generated story. Our proposed model under controllable conditions can generate stories with multiple fine-grained psychological states of multiple characters by assigning emotional change lines to the characters in the story. We design a Character Psychological State Controller (CPSC). Each time step in the decoder selects a character who will be described at the current time step and corresponding psychological state we assigned manually from many characters in the story. The selected character’s psychological state will then be controlled and determined. For generating coherent stories easily, we introduce the external knowledge that can facilitate language understanding and generation. ConceptNet is a commonsense semantic network that consists of triples with head, relation, and tail, which can be represented as $g = (h,
The head and tail can be connected by their relation, where we apply this method to build a bridge of story context and next story sentence. Inspired by the Graph Attention [11], we design a knowledge-enhanced method to represent the knowledge triples. The method treats the knowledge triples as a graph, from which we can better interpret the semantics of an entity from its neighboring entities and relations. For the reason of the particularity of our model, the evaluation metric of the accuracy of psychological state control is investigated.

Our contributions are as follows:

(i) We develop three psychological theories as controllable conditions, which are used to describe characters in the stories

(ii) To enhance the semantic and richness of the story, we introduce the external knowledge graph into the generation model

(iii) We propose a model K-GuCCI, which utilizes external knowledge to enhance story generation’s coherence while ensuring the controllability of conditions. We design a character psychological state controller that achieves fine-grained psychological state control of the characters in the story

(iv) We explore a novel evaluation metric for the accuracy rate of psychological state control

(v) The experimental results demonstrate superior performance in various evaluating indicators and can generate more vivid and coherent stories with fine-grained psychological states of multiple characters. We also verify the effectiveness of the designed modules

2. Related Work

2.1. Text Generation with External Knowledge. Introducing external knowledge to natural language tasks is a trend in recent several years. Semantic information can be enhanced by external knowledge to help complete many works, particularly important in story generation. Chen et al. [12] utilize external knowledge to enhance neural data-to-text models. Relevant external knowledge can be attended by the model to improve text generation. Wang et al. [13] introduce the knowledge base question answering (KBQA) task into dialogue generation, which facilitates the utterance understanding and factual knowledge selection. Zhou et al. [11] first attempt to use large-scale commonsense knowledge in conversation generation. They design a graph attention mechanism in encoder and decoder, which augments the semantic information and facilitates a better generation. Guan et al. [14] focus on generating coherent and reasonable story endings by using an incremental encoding scheme. All of the above works show the effectiveness of introducing external knowledge. In our work, the proposed model K-GuCCI mainly focuses on the characters’ psychological state.

Figure 1: We take the psychological states consisting of Maslow, Reiss, and Plutchik three theories as our controllable conditions. The pyramid in (a) shows the five categories of Maslow from the physiological to the spiritual level. The right ladder represents the Reiss motives as a subset of Maslow. Plutchik’s wheel of emotions in (b) represents the eight basic dimensions of emotions.

Table 1: An example of our story generation. The input sentence is the beginning of the story. “Jervis” and “girlfriend” are the characters in the story. We have three psychological theories, like “Plutchik,” “Maslow,” and “Reiss.” For Plutchik, each character has an emotional change line assigned manually. The characters will share identical motivations based on Maslow and Reiss. The following table shows an example of the story.

| Input      | Jervis has been single for a long time. |
| Characters | Jervis, girlfriend                     |
| Girlfriend | None–none–joy–joy–joy                   |
| Maslow     | Love                                    |
| Reiss      | Contact, romance                        |

Table 1: An example of our story generation. The input sentence is the beginning of the story. “Jervis” and “girlfriend” are the characters in the story. We have three psychological theories, like “Plutchik,” “Maslow,” and “Reiss.” For Plutchik, each character has an emotional change line assigned manually. The characters will share identical motivations based on Maslow and Reiss. The following table shows an example of the story.

| Input      | Jervis has been single for a long time. |
| Characters | Jervis, girlfriend                     |
| Girlfriend | None–none–joy–joy–joy                   |
| Maslow     | Love                                    |
| Reiss      | Contact, romance                        |

| Input      | Jervis has been single for a long time. |
| Characters | Jervis, girlfriend                     |
| Girlfriend | None–none–joy–joy–joy                   |
| Maslow     | Love                                    |
| Reiss      | Contact, romance                        |
2.2. Story Generation under Conditions. Story generation has attracted much attention recently. Jain et al. [15] leverage a sequence-to-sequence recurrent neural network architecture to generate a coherent story from independent descriptions. The standalone textual descriptions describing a scene or event are converted to human-like coherent summaries. Fan et al. [2] explore coarse-to-fine models that first generate sequences of predicates and arguments conditioned upon the prompt and then generate a story with placeholder entities. Finally, the placeholders are replaced with specific references. Fan et al. [1] propose a hierarchical model that can build coherent and fluent passages of text about a topic. Yu et al. [16] propose a multi-pass hierarchical CVAE generation model, targeting to enhance the quality of the generated story, including wording diversity and content consistency. A lot of emotional text generation tasks have emerged in recent years. Xing et al. [17] use a sequence-to-sequence structure with topic information to produce exciting chatbot responses with rich information. Ghosh et al. [18] generate conversational text conditioned on affect categories, which customize the degree of emotional content in generated sentences through an additional design parameter. There are also some generation tasks with emotion or sentiment [4, 19, 20]. They only use a specific or binary emotion like positive or negative to express emotion or sentiment. Unlike the above works, we aim to generate a story with different characters’ psychological states change, including multiple motivations and emotions. We use [21] as our dataset that composes of a five-sentence story. The characters’ motivations and emotions in the story will change with the development of the story plot. Paul and Frank [22] also use the dataset to do a sentiment classification task according to the psychological state.

3. Methodology

Figure 2 shows an overview of the K-GuCCI model. The proposed model can generate vivid and coherent stories under controllable conditions. We assign the multiple fine-grained psychological states of characters as controllable conditions. We perform story generation using a Seq2Seq structure [23] with external knowledge using graph attention method, where BiLSTM [24] and LSTM [25] are used as encoder and decoder, respectively. We design a Character Psychological State Controller (CPSC) module to control each character’s fine-grained psychological state.

3.1. Problem Formulation. Formally, the input is a text sequence \( X = (x_1, \cdots, x_n) \) that is a beginning of the story, which consists of \( n \) words. We also take story context \( C = (x_1^c, \cdots, x_m^c) \) consisting of \( m \) words as input to increase the coherence of the story, which represents historical story sentences of the input sentence \( X \). We represent the external knowledge as \( K = ([h_1, r_1, t_1], \cdots, [h_l, r_l, t_l]) \), where \([h, r, t] \) is the triple consisting of head, relation, and tail. Meanwhile, we quantify a psychological state score of each character for three theories Plutchik, Maslow, and Reiss: \( S_{m}^{\text{emr}} = (S_{p}^p, S_{m}^m, S_{r}^r), \cdots, (S_{p}^p, S_{m}^m, S_{r}^r) \), where \( j \) is the characters’ number in the story. The output target is another text sequence \( Y = (y_1, \cdots, y_k) \) that consists of \( k \) words. The task is then formulated as calculating the conditional probability \( P(Y \mid X, C, PMR, K) \), where \( PMR \) represents the psychological state.

3.2. Character Psychological State Controller. The Character Psychological State Controller is used to control which and how much characters’ psychological state can be used to describe the story. For psychological state, because it is composed of multiple psychological states, we quantify the psychological state so that it can be accepted by the model.

3.2.1. Psychological State Representation. We quantify a psychological state as a PMR matrix that is used to describe the fine-grained psychological state of characters in the story. As shown in Figure 3, we just display Plutchik scores of each character for the third sentence in the story, where the score “0” denotes no current emotion. The higher the score, the richer the current emotion. We normalize these scores and then build a vector for each emotion or motivation. We set the characters number as maximum \( n \) for processing different characters in the stories. We concatenate them as multiple characters score matrix, i.e., Plutchik score \( S_{p}^{m} \), Maslow score \( S_{m}^{m} \), and Reiss score \( S_{r}^{r} \). Then, a word vector matrix for the three psychological states is randomly initialized as \( V_{p}^{p} \), \( V_{m}^{m} \), and \( V_{r}^{r} \), respectively. Figure 3 shows the Plutchik score matrix \( S_{p}^{m} \) and word vector matrix \( V_{p}^{p} \). For the Plutchik score matrix \( S_{p}^{m} \), we pad the matrix with less than the maximum number of characters. Each row represents a character, and each column represents a score for each emotion. For word vector matrix \( V_{p}^{p} \), each row expresses a representation of an emotion. The word vector matrix will be multiplied by the characters score matrix; then, the product will be mapped into a low dimension space. We obtain the Plutchik matrix, the Maslow matrix, and the Reiss matrix subsequently. The formulation is as follows:

\[
P_i = W_p (S_{p}^{m} \times V_{p}^{p}) + b_p, \tag{1}
\]

\[
M_i = W_m (S_{m}^{m} \times V_{i}^{m}) + b_m, \tag{2}
\]
Story: Gina wanted a unicorn folder like her friend Tami. She had never seen anything like it. She had already been in trouble for talking while the teacher was. So she decide to wait till the teacher was done teaching. Once the teacher finished she asked Tami about the folder.

Character: Teacher, Gina, Tami

<table>
<thead>
<tr>
<th>Character</th>
<th>Joy</th>
<th>Trust</th>
<th>Fear</th>
<th>Surprise</th>
<th>Sadness</th>
<th>Disgust</th>
<th>Anger</th>
<th>Anticipation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teacher</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>0</td>
<td>6</td>
<td>8</td>
<td>0</td>
</tr>
<tr>
<td>Gina</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>4</td>
<td>0</td>
</tr>
<tr>
<td>Tami</td>
<td>0</td>
<td>4</td>
<td>7</td>
<td>0</td>
<td>4</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Joy</th>
<th>Trust</th>
<th>Fear</th>
<th>Surprise</th>
<th>Sadness</th>
<th>Disgust</th>
<th>Anger</th>
<th>Anticipation</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0.6</td>
<td>0.7</td>
<td>1</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0.6</td>
<td>1</td>
<td>0</td>
<td>0.6</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td>...</td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Figure 3: An example of constructing the PMR matrix. We just attend to the third sentence in blue in the story. There are three characters in this story. The table above shows the Plutchik score for each character in the dataset. The matrix in orange and in green displays the Plutchik and Plutchik word vector matrix $V_p$, respectively.

$$R_i = W_i(S_i' \times V_i') + b_i,$$  \hspace{1cm} (3)

where $W_p$, $W_m$, and $W$ are the weight matrices. $b_p$, $b_m$, and $b_i$ indicate the biases, and $i$ is the $i$-th character. The Plutchick, Maslow, and Reiss matrices will be concatenated as characters PMR matrix for the convenience of calculation:

$$PMR_i = W_{pmr}(S_{pmr}^{i} \times V_{pmr}^{i}) + b_{pmr},$$  \hspace{1cm} (4)

3.2.2. Controllable Psychological State. We control the multiple characters’ psychological states by first selecting a character who will be described at each decoder time step, and then, the selected character’s psychological state will be controlled using an attention method.

At each step $t$ of decoder, we use a feed-forward layer to compute a character gate vector $g_{t}^{\text{char}}$. The softmax activation is used to calculate a probability distribution of characters in the story; then, a one-hot mechanism picks up a character with maximum probability $o_{t}^{\text{char}}$. We multiply the PMR, with the $o_{t}^{\text{char}}$ to obtain the character’s psychological states:

$$g_{t}^{\text{char}} = \text{softmax} \left( W_g y_{t-1} ; h_{t-1} ; c_i \right),$$  \hspace{1cm} (5)

$$o_{t}^{\text{char}} = \text{one-hot} \left( g_{t}^{\text{char}} \right),$$  \hspace{1cm} (6)

$$s_{t}^{\text{char}} = o_{t}^{\text{char}} \times PMR_i,$$  \hspace{1cm} (7)

where $W_g$ is the weight matrix, $y_{t-1}$ is the input word, $h_{t-1}$ is the decoder hidden state, and $c_i$ is the context vector. After that, we calculate a psychological state vector $c_{t}^{PMR}$ at step $t$ which is taken as the final condition to control model generation.

$$e_{t,i} = V_a^T \tanh \left( W_a o_{t}^{\text{char}} + U_a h_{PMR,i} \right),$$  \hspace{1cm} (8)

$$\alpha_{t,i} = \frac{\exp \left( e_{t,i} \right)}{\sum_{i=1}^{n} \exp \left( e_{t,i} \right)},$$  \hspace{1cm} (9)

$$c_{t}^{PMR} = \sum_{i=1}^{n} \alpha_{t,i} PMR_i,$$  \hspace{1cm} (10)

where $W_a$ and $U_a$ are the weight matrices, $t$ is the time step, $i$ is the $i$-th character, and $c$ is the number of characters.

3.3. Knowledge-Enhanced Generation Model

3.3.1. Knowledge Encoding. In order to represent a word more meaningful and a story more coherent, we use knowledge aware representation and attention based on the context to enhance the semantic expression in the encoder. We first calculate a knowledge graph vector $c_{kg}$ which attends to the triple of the words in the knowledge graph, and then a context vector $c_{con}$ to attend to the context information; both of which are as the input with the sentence together. We get a knowledge graph vector $c_{kg}$ by using graph attention [11]. The words in the sentences have their own knowledge representation by triples. In this way, the words can be enriched by their adjacent nodes and their relations. For a context vector
where \( h^{(i)} \) is the hidden state of the \( i \)-th sentence of the story. \( e^{(i)}_c \) is the concatenation of the knowledge graph vector \( e^{(i)}_{kg} \) and context vector \( e^{(i)}_{con} \).

\[
e^{(i)}_c = [e^{(i)}_{kg} ; e^{(i)}_{con}],
\]

where \( e^{(i)}_{con} \) is the context attention vector of \( i \)-th sentence. \( e^{(i)}_{kg} \) is the knowledge graph vector of the \( i \)-th sentence and is formulated as follows:

\[
e^{(i)}_k = h^{(i)} W_k g(x)^{(i-1)},
\]

\[
\alpha^{(i)}_k = \frac{\exp (e^{(i)}_k)}{\sum_{i=1}^{k} \exp (e^{(i)}_k)},
\]

\[
e^{(i)}_{kg} = \sum_{i=1}^{k} \alpha^{(i)}_k g(x)^{(i-1)},
\]

where \( g(x)^{(i-1)} \) is the graph attention vector in [11]. The whole story generation process will always be followed by the knowledge graph vector and context vector, which is the soul that keeps the story coherent.

3.3.2. Incorporating the Knowledge. We concatenate the last time step word embedding vector \( e(y_{t-1}) \), PMR context \( c^{PMR} \), knowledge graph vector \( e^{(i)}_k \), and attention context \( c_i \), which represent incorporating the external knowledge and psychological state into the generation model. The LSTM hidden state is updated as follows:

\[
h_t = \text{LSTM} \left( [e(y_{t-1}) ; e^{PMR}_t ; e^{kg}_t ; c_i], h_{t-1} \right).
\]

We minimize the negative log-likelihood objective function to generate expected sentences.

\[
L_{\text{NLL}} = -\frac{1}{N} \sum_{i=1}^{N} \sum_{t=1}^{T} \log p \left( y^{(i)}_t | y^{(i)}_{<t}, X^{(i)}_t, C^{(i)}, PMR^{(i)}, k^{(i)} \right),
\]

where \( N \) is the story number in the dataset, and \( T \) is the time step of the \( i \)-th generated sentence in the decoder. \( X^{(i)}_t \) represents the \( i \)-th sentence in the dataset. Similarly, \( C^{(i)}, PMR^{(i)} \), and \( K^{(i)} \) represent the \( i \)-th context, \( i \)-th PMR matrix, and \( i \)-th knowledge triples in the dataset, respectively.

4. Experiments

4.1. Dataset. The dataset in [21] is chosen as our story corpus, consisting of 4k five-sentence stories. The corpus contains stories where each sentence is not only annotated but also with characters and three psychological theories. Figure 4 displays the statistic of the psychological states. Plutchik’s emotion appears more frequently than Maslow’s and Reiss’s motivation. Particularly, “joy” and “participation” are most in the Plutchik states. The Reiss categories are subcategories of the Maslow categories. We use different methods to process the dataset for Plutchik, Maslow, and Reiss. Three workers who are employed by the original author annotate the original data. Intuitively, the workers will have different viewpoints, so we sum up the Plutchik scores and normalize them. Maslow and Reiss have no repeated words; thus, we use a one-hot vector to represent them. We split the data as 80% for training and 20% for testing. In the test phase, we input the story’s first sentence and the normalized psychological states scores. Table 2 statistics the character number in each story sentence. They are most in the range 1-3, and the largest character number is 6. Thus, we set the character number as 3.

4.2. Baselines. We compare our model with representative baselines to investigate the effectiveness of the K-GuCCI model. The baselines are as follows:

Seq2Seq model introduced by Google in 2014 and has encoder, decoder, and intermediate step as its main components. The model can map input text with fixed length to
output text with fixed length. It is widely used in text generation tasks. Our model is improved based on Seq2Seq. Therefore, the Seq2Seq baseline can be used to compare to prove our model’s effect on emotional controllability and fluency.

Inc-S2S is an incremental Seq2Seq model mentioned in [3]. Different from the implementation in [3], we incorporate the psychological states into the model. The story sentences are generated according to the beginning of the story sentence and context. Compared with the Inc-S2S model, the effectiveness of the Character Psychological State Controller can be proved.

Transformer [27] is a novel architecture that aims at solving natural language processing tasks while handling long-range dependencies with ease. Since the Transformer model facilitates more parallelization during training, it has led to the development of pretrained models such as BERT [28], GPT-2 [29], and Transformer-xl [30], which have been trained with huge general language datasets.

GPT-2 [29] shows an impressive ability to write coherent and passionate essays. Its architecture is composed of the decoder-only transformer and it can be trained on a massive dataset. There are many works in natural language generation tasks that use GPT-2-based model.

SoCP [10] proposes a novel model called SoCP, which can generate a story according to the characters’ psychological states. The model is most relative to us. Different from that, our model introduces a knowledge graph to enhance semantic information and promote the coherence of the story.

4.3. Experimental Settings. Based on the above, we fix the character number to three. If the character number is smaller than three, use "none" as a character. The pretrained glove 300 dimension vector is used as our word embedding vector. We map the PMR matrix from a high dimension to a 256 low dimension. We implement the encoder as a two-layer bidirectional LSTM and the decoder as a one-layer LSTM with a 256 hidden size. The batch size is 8, and 0.2 is the dropout [31]. The learning rate of Adam optimizer [32] is initialized as 0.0003.

4.4. Evaluation Metrics. BLEU [33] is a metric to quantify the effectiveness of generated text according to compare a candidate sentence of the text to one or more reference label sentences. Although designed for translation, it is commonly utilized for a suite of natural language processing tasks.

ROUGE, stands for Recall Oriented Understudy for Gisting Evaluation, is a set of metrics used for evaluating the automatic text summarization and machine translations. The metrics basically compare the similarity between generated sentences and reference sentences.

METEOR is based on the harmonic mean of unigram accuracy and recall, weighted higher than accuracy with recall. In the more common BLEU metric, the metric will correct some of the issues and also produce a strong correlation with human judgement at the level of the sentence or section.

In AGPS (Accuracy of Generated Psychological State), we pretrain a classifier to evaluate the accuracy of the generated psychological state. There are many approaches to train a classifier [34–36]. We utilize bidirectional LSTM to pretrain a classifier to classify the generated sentence like sentiment classification. This demonstrates our model’s capacity to convey emotions. The name of the character Char and the sentence X as input are concatenated. In this fashion, several training pairs with different outputs in similar sentences can be obtained when different characters in a sentence have various psychological states. The compact vector $h_{c}^{ij}$ can be accessed by BiLSTM, and then, we utilize two feed-forward layers to compact it into the output size.

$$h_{c}^{ij} = \text{BiLSTM}([\text{Char} ; X]),$$

$$p = W_{p}\left(\text{Relu}\left(W_{o}\left(h_{c}^{ij}\right) + b_{o}\right) + b_{p}\right).$$

4.5. Result Analysis. We have the Seq2Seq framework, the Inc-S2S, and Transformer as the baseline model. We use automatic assessment metrics, including the proposed metric AGPS, to compare our model with baseline models’ effectiveness. The experiments can prove our components and demonstrate the consistency in which the generated sentence psychological states correspond with our previous collection. All matrix’ scores in our model are the highest, as seen in Table 3, which shows the effect of our built modules, and the generated sentences are coherent.

We see that Seq2Seq has better results for BLEU, ROUGE, and METEOR than the Transformer framework.

Table 3: Automatic evaluations of the proposed model and the baseline models. The context-merge and context-independent represent different methods of encoder mentioned in [10]. The former is to encode the context and sentence together, while the latter encodes them separately and then concatenates them.

<table>
<thead>
<tr>
<th>Model</th>
<th>BLEU</th>
<th>ROUGE</th>
<th>METEOR</th>
<th>AGPS</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>B-1</td>
<td>B-2</td>
<td>B-3</td>
<td>B-4</td>
</tr>
<tr>
<td>Seq2Seq</td>
<td>0.202</td>
<td>0.049</td>
<td>0.016</td>
<td>0.007</td>
</tr>
<tr>
<td>Transformer</td>
<td>0.207</td>
<td>0.053</td>
<td>0.020</td>
<td>0.008</td>
</tr>
<tr>
<td>Inc-S2S</td>
<td>0.224</td>
<td>0.053</td>
<td>0.017</td>
<td>0.006</td>
</tr>
<tr>
<td>SoCP+context-merge</td>
<td>0.216</td>
<td>0.056</td>
<td>0.021</td>
<td>0.010</td>
</tr>
<tr>
<td>SoCP+context-independent</td>
<td>0.232</td>
<td>0.062</td>
<td>0.025</td>
<td>0.011</td>
</tr>
<tr>
<td>(Ours)K-GuCCI</td>
<td>0.242</td>
<td>0.070</td>
<td>0.029</td>
<td>0.014</td>
</tr>
</tbody>
</table>
Figure 5: Continued.
The reason may be that the Seq2Seq model is more appropriate for short texts than Transformer. In addition, the outcomes of our method are better than the SoCP with context-merge method and context-independent method. The effectiveness of external knowledge that can enrich the generated stories is also reflected in this table. Our proposed model, of all the ones, has the best efficiency. The training speed of the Transformer, however, is much higher than that of Seq2Seq. It reflects the benefit of the Transformer in training speed because of the parallelism of operation.

We design AGPS to assess whether the emotion of the generated sentences is consistent with the settings. We intuitively assume that the score without input emotion will be lower. The performance of Inc-s2s is between our model and other models, which shows that our model performs efficiently for our built components.

The result of the K-Gucci model is better than that of the SoCP model, which shows that the story can be enriched by introducing knowledge.

4.6. Model Effect Analysis

4.6.1. The Effect of the Character Psychological State Controller. We display the attention weight distribution to demonstrate the relation between the generated sentences and psychological states. As seen in Figure 5, the model provides interpretability dependent on the character’s psychological state controller. The brighter the square corresponding to the two words while generating the next word is, the stronger the relationship between them will be. Visualization of the focus maps offers a proof of the model’s ability to recognize which psychological state corresponds to the character. A word may have many different color squares, suggesting that our component can read several characters’ psychological states automatically and can select the psychological states for the character automatically. The black square suggests that no responsible psychological state is reached because the feeling is not actually conveyed by all words, such as “a” and “the”. The model correctly chooses the elements from the psychological states in the examples displayed. The first example focuses on the emotions of Plutchik, such as “fear” and “anger,” while the second example discusses Maslow and Reiss’s elements, such as “spiritual growth” and “indep.” The term “hospital” is correlated with the Plutchik in the third attention diagram, such as “fear,” “surprise,” and “sadness,” implying that “hospital” is typically associated with a character’s negative emotion. In the fourth attention diagram, the word “however” predicts a vital turning point and negative outcomes that the character will be failed in the exam, which is also compatible with the “depression” and “anger” emotions.

4.6.2. The Effect of the External Knowledge. As seen in Table 4, the evaluation matrix shows that the performance of our model is better than other models. In addition, K-Gucci demonstrates the effect of external knowledge in Table 4. For example, “necklace” is linked to “like it,” and “losing his mind” is linked to “go to the hospital,” which illustrates that the generated stories are coherent and reasonable. In the meantime, the conditions we set can control the stories while the coherence of the story is assured. With our setting, as we assigned the Plutchik emotional lines, the emotions of the stories will shift.

4.7. Case Study

4.7.1. Comparison with Baseline Models. The examples of stories generated by our model and the baseline models are shown in Table 4. A coherent story can be constructed by the psychological state condition we have given.

We see that it generates typically repetitive sentences by the Seq2Seq model, and it cannot accept all the characters. In Table 4, example 1 produced by the baseline model describes only one character “Jane,” but can generate
friends by the K-Gucci. In example 2, we see that with our defined psychological state condition, the baseline model cannot vary the story and even have incorrect feelings, but our K-Gucci model can match it correctly. The GPT-2 model is capable of generating rational phrases but has several repetitions. Overall, by manipulating the feelings of the characters, our proposed model will generate good stories. There are also some stories that are not coherent, so it is still a challenge for us.

4.7.2. Controllability. The examples in Table 5 show the controllability of generated stories under different psychological state conditions. The first example in Table 5 compares the generated stories under various condition scores using an identical Plutchik element. In specific, we set the Plutchik “joy” with different scores in the first example. Some obvious terms such as “great,” “excited,” or “really liked” are produced when the score sets 1. As the “joy” score gets lower and lower, the terms produced get more and more negative. When the score is set to 0, some negative terms, such as “nervous” or “not good” are produced. The second example shows the produced stories with various indicators from Plutchik. We assign various Plutchik indicators to “surprise,” “fear,” and “anger.” It produces several words, such as “was surprised” or “shocked” when Plutchik is “surprise.” When Plutchik is “fear,” the words “was afraid of” or “scared” are formed. The term “angry” can be formed when Plutchik is “anger.” In the third case, for the multiple Plutchik metrics, separate scores are assigned. In the third case, in the produced stories, several emotions are portrayed.

The above examples demonstrate the controllability of our model. On the other hand, in the examples mentioned above, several incoherent stories tell us that although the
model performs well in emotion control, it still needs to be improved in coherence.

5. Conclusion

Traditional story generation models can only generate stories with one specific emotion and lack coherence. In this paper, we propose a model called K-GuCCI, which can generate more vivid and coherent stories under controllable conditions. We take the three psychological state theories as our controllable conditions and design a character psychological state controller, which controls the psychological state of multiple characters in the story. We introduce the external knowledge graph to enhance the semantic and richness of stories. In addition, we design an evaluation metric called AGPS to evaluate the accuracy of the generated psychological state. For future work, we will use an advanced pretrained model to generate more coherent texts. In the field of wireless communications and mobile computing, there are many applications of the recommender system, such as [37–39], and the Internet technology, such as [40–42]. We want to use our method to recognize users’ emotions, generate high-quality text, and serve more Internet applications.

Data Availability

The data that support the findings of this study are available at https://uwnlp.github.io/storycommonsense/.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References


