RST Discourse Structure Improves Story Ending Generation

1 Introduction

Story ending generation is a crucial task for automatic story generation, which completes a story with a reasonable ending sentence based on given contexts. A typical example can be found as follows:

Context | Gina misplaced her phone at her grandparents. It wasn’t anywhere in the living room. She realized she was in the car before. She grabbed her dad’s keys and ran outside.

Ending | She found her phone in the car.

Conventionally, an encoder-decoder sequence generation model is applied for predicting the ending sentence. Here, the encoder turns the source sentences into fixed-length vectors, then the decoder produces the resultant sequence with the help of attention mechanism.

Two existing works can be found in literature on this task. Each work puts an unique assumption on the contextual structure of sentences in a story. The most common modelling approach [14, 5] is to concatenate the contextual sentences, forming a long sequence and encode it. In this approach, all contextual sentences can be directly accessed by the decoder. Therefore, the sentences are considered equally important. To capture the order and the relationships between adjacent sentences, Guan et al. [4] represents context clues by incrementally encoding contextual sentences from left to right to build a context vector. The decoder then observes the context vector containing information from all previous sentences. In this approach, latter sentences are considered as more important. Fig. 1a shows the implicit discourse structure in this case.

Story Cloze Test is a task that given contextual sentences, one has to select one correct ending from given candidates. Srinivasan et al. [11] found that the model using only the last sentence in the context achieves comparable performance with the models using full context.

To investigate whether their finding is also appropriate in story ending generation, we conducted a preliminary experiment, which will be detailed in Section 5.2. As a result, we find that the first sentence almost always contain topical or background information helpful for generating a better ending. Based on the preliminary results, we consider the topic sentence as the most important context.

Therefore, to generate more coherent endings, we propose to find the importance of each sentence in the context. Discourse parsing is such a promising method to automatically uncover coherence structures (i.e., trees) for multiple sentences. It has been successfully applied in down-stream tasks such as question answering [1], and automatic essay scoring [8]. Rhetorical Structure Theory (RST) [7] is one of the most widely accepted theories of discourse structure. Figure 1b illustrates an encoding strategy according to RST-based discourse structure. The hierarchy of sentences represents their relative importance [6], which we consider to be useful for story ending generation.

Our contribution can be summarized as two folds. First, we propose a simple and efficient encoding strategy that extends Transformer [12] to exploit RST-based discourse structures for encoding contextual sentences in a story. Second, we perform em-
2 Related Work

Story Ending Generation is the task to complete a story, which has been approached by several neural based methods in the recent years. The ROCstory corpus often used in this task was designed for Story Cloze Test [9]. In this test, the model needs to understand the logic, find clues and compare two candidates. However, story ending generation is far more challenging than making choices because the model needs to generate both resonable and fluent sentence.

Early attempt [14] relied on reinforcement learning method on copy mechanism in order to enhance the probability of selecting the word from the context. Recently, Gupta et al. [5] aimed to generate more interesting sentences with different conditions including sentiment, length and keyphrase. Guan et al. [4] represented context clues by incremental encoding, and leverages common-sense knowledge by multi-source attention to effectively capture the coherence and logic of story.

3 Rhetorical Structure Theory

Rhetorical Structure Theory (RST) [7] assumes that a tree structure can be derived for coherent text. In RST, leaf nodes are clauses-level non-overlapping text segments (or elementary discourse units, EDUs), and consecutive text segments are combined by rhetorical relations recursively to form lager segments. The rhetorical relations have two components: Nucleus and Satellite. A nucleus corresponds to the more salient segment of the relation, while a satellite corresponds to more supporting one.

In this paper, we employ RST discourse dependency structures [13], such as the one in Figure 2, for story ending generation. To obtain RST parses for context stories in our experiments, we used an off-the-shelf RST parser [3].

4 Proposed Encoder Model

We propose a novel context encoding strategy based on RST discourse dependency graphs for story ending generation. Fig. 3 gives an overview of the encoder model architecture, which contains two components. The first component is the original word-level Transformer encoder. The second component applies K layers self-attention mechanism for encoding sentence-level information according to a specific tree structure.

Word-Level Encoder In this paper, we use $S_1$, $S_2$, $S_3$ to denote contextual sentences, where each sentence $S_i$ may have different number of tokens. In the first stage, a regular Transformer encoder is applied to encode the words in each sentence into word-level feature vectors as shown in Fig. 3 as the first layer of vectors.

Sentence-Level Encoder Similar to BERT [2], we use the word feature corresponding to the start token (bos) as the feature vector for each sentence. Now, we have $N$ sentence-level feature vectors. Next, we feed these sentence vectors into K self-attention layers. We apply binary masks constructed according to the tree structure.

In detail, when computing the weighted summarization in the attention, each $S_i$ is assigned a mask $m_i \in \{0, 1\}^N$. Each mask is an N-dimensional binary vector. An example of $m_i$ can be $[1, 0, 0, 0]^T$. The value $m_{ij}$ is 1 when $S_j$ depends on $S_i$ based on the RST discourse dependency tree. For the example tree in Fig. 2, all mask $m = m_1, m_2, \ldots, m_N$ forms

\[
m = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}
\]

(1)

When applying self-attention for $K$ iterations, we denote the hidden state for $S_i$ in $k$-th iteration as $h_i^k$. In each iteration, the hidden state $h_i^k$ is updated as the following weighted summarization:

\[
h_i^{k+1} = \sum_{j=1}^N w_{ij}^k h_j^k
\]

(2)

\[
w_{ij}^k = \frac{\exp(f(h_i^{k-1}, h_j^{k-1}))}{\sum_{j=1}^N \exp(f(h_i^{k-1}, h_j^{k-1}))}
\]

(3)

\[
f(h_i^{k}, h_j^{k}) = \begin{cases} \text{Score}(h_i^{k}, h_j^{k}), & \text{if } m_{ij} = 1 \\ -\infty, & \text{otherwise} \end{cases}
\]

(4)

where $w_{ij}^k$ is the attention weight that matches $S_j$ to $S_i$ in the $k$-th iteration. The score function $\text{Score}(\cdot)$
This founding confirms that to generate a coherence ending, the topic sentence is necessary so that RST discourse dependency structure is valuable in this task because the head in the RST discourse dependency structure is always the topic sentence.

5.3 Baselines

In this experiment, we set several baselines including the original Transformer and our proposed model with different tree structures.

- **Transformer (T):** Sentences are concatenated into a long sentence and trained by regular Transformer.
- **Transformer* (T*):** With the aforementioned model, all the masks are set to 1, thus each sentence contributes equally to the model.
- **Transformer Random (T (rand)):** In order to prove that the RST structure is helpful, we assign random mask to each context.
- **Transformer Last (T (last)):** Similar to the previous work [4], each sentence is encoded to its next sentence.
- **Transformer First (T (first)):** The sentences all connect to the first sentence based on the expectation that the first sentence always provide topic, which helps generating more coherent ending.
- **Transformer RST (T (RST)):** The model is equipped with RST discourse dependency structure which is our proposal.

5.4 Results

We conduct the automatic evaluation on the 1,874 stories in the test dataset. We adopt BLEU, METEOR, ROUGE-L, CIDEr and perplexity (PPL) to evaluate the generation performance. Except for the Perplexity score, higher scores are better. To further evaluate the quality of generated stories, Sagarkar et al. [10] proposed to evaluate overall, relevance and interestingness score with trained models. We also adopt their models to evaluate our stories.

As shown in Table 1, our RST based transformer model achieves the best score on perplexity (PPL), which means our proposed model assigns higher probability to the gold sentence. Moreover, we achieve a 0.67% improvement from the second place in the CIDEr score which means this model contains more consensus of how people writes the ending. In the metrics evaluated by Sagarkar’s model, we get the best performance on overall and relevance.

Table 3 shows examples of the generated ending with different models. Please note that as both T(first) and T(RST) consider the first sentence as root, they generate similar sentences. The results reveals that using different tree structure can indeed influence the output.
Table 1: Automatic and neural based (O, R, I) results. O, R, and I respectively denote Overall score, Relevance score and Interestingness score [10].

Table 2: Automatic results in the preliminary experiment. S denotes the index of the sentence.

Table 3: Examples generated by different models.

6 Conclusion

In this paper, we leverage the RST discourse structure for story ending generation task. The results show that the RST discourse structure helps the model to capture the importance of sentences. However, several limitations shall be concerned. First, due to that RST was not originally designed based on the story structure, ideally, we need to develop new relation labels and structure theories to fit the task. Second, according to recent works, models are weakened when facing longer stories. We conjecture that in the long story generation, RST structure further helps the model to understand the context. We leave these points for our future work. 

Acknowledgements

This paper is based on results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO).

References