Improving Chinese Story Generation via Awareness of Syntactic Dependencies and Semantics

Henglin Huang1; Chen Tang1; Tyler Loakman2, Frank Guerin1 and ChenghuaLin2†
1Department of Computer Science, The University of Surrey, UK
2Department of Computer Science, The University of Sheffield, UK
{hh01034,chen.tang,f.guerin}@surrey.ac.uk
{tcloakman1,c.lin}@sheffield.ac.uk

Abstract

Story generation aims to generate a long narrative conditioned on a given input. In spite of the success of prior works with the application of pre-trained models, current neural models for Chinese stories still struggle to generate high-quality long text narratives. We hypothesise that this stems from ambiguity in syntactically parsing the Chinese language, which does not have explicit delimiters for word segmentation. Consequently, neural models suffer from the inefficient capturing of features in Chinese narratives. In this paper, we present a new generation framework that enhances the feature capturing mechanism by informing the generation model of dependencies between words and additionally augmenting the semantic representation learning through synonym denoising training. We conduct a range of experiments, and the results demonstrate that our framework outperforms the state-of-the-art Chinese generation models on all evaluation metrics, demonstrating the benefits of enhanced dependency and semantic representation learning.

1 Introduction

Story generation presents a challenging task, primarily due to the difficulty that end-to-end neural models experience in maintaining logical coherence during long text generation (Tang et al., 2022). These challenges are even more prominent for the task of story generation in Chinese, due to increased levels of ambiguity stemming from the absence of explicit delimiters for character separation (Tian et al., 2020). Recent works, on the other hand, have aimed to improve long text generation through the proposal of more efficient frameworks (Rashkin et al., 2020a; Goldfarb-Tarrant et al., 2020; Brahman and Chaturvedi, 2020), or through augmenting existing frameworks with pre-training and the injection of extra knowledge (Xu et al., 2020; Guan et al., 2020, 2022).

However, we observe that current generation models still struggle to generate fluent and coherent Chinese stories, which may be the result of the inefficient capturing of features in written Chinese. For example, Chinese characters have a range of morphological parsing strategies, e.g., “小心地滑” can be understood as “小心地滑” (caution wet floor) or “小心地滑” (carefully slide), whose meaning is highly dependent on context (Chen et al., 2018; Li et al., 2022). This may cause important sentential roles such as subjects, predicates, and objects to be difficult to identify and process by neural models. Additionally, when neural networks learn the semantics of an utterance, synonymous expressions may lead to confusion, damaging the robustness of the generation model, e.g., “游历”, “周游”, and “游览” are different Chinese words but all express ”travelling” in Chinese. We therefore propose to train neural networks to learn the semantic-level features contained in context, rather than the low-level features of characters.

To this end, we propose a novel data augmented story generation framework illustrated in Figure 1, including a LongLM (Guan et al., 2022) based con-
We employ HanLP (He and Choi, 2021) to parse our story generation task based on related subjects and objects via dependency parsing, all of which are essential in expressing the event represented within a sentence. The semantic denoising module, based on SimBert\(^2\) (Su, 2020), generates a range of different, yet essentially synonymous sentences, to force the neural network learn the semantic representations of key entities and different expressions. Overall, our proposed framework enhances the ability for language understanding in written Chinese via training to capture the dependencies and semantics contained within sentences, in order to then generate stories.

We conduct a range of experiments on the latest public benchmark for Chinese story generation (Guan et al., 2022), and the results indicate that the model trained with our framework substantially outperforms the state-of-the-art (SOTA) baselines on all metrics.\(^3\) This indicates that our framework improves the generated stories via enhanced capturing of syntactic dependencies and semantic features.

2 Methodology

We formulate our story generation task based on the OutGen task from LOT (Guan et al., 2022), a Chinese story generation benchmark. The definition of the task is: An outline \(X\), which contains an unordered list of an arbitrary number of Chinese phrases concerning characters and events, is given as the input. The model is required to generate a coherent story \(Y = \{y_1, y_2, \ldots, y_n\}\) where \(y_i\) denotes the \(i\)-th token (Chinese character) in the story.

2.1 Dependency Tagging

We employ HanLP (He and Choi, 2021) to parse dependencies within Chinese stories. Unlike in English, the basic unit of Chinese dependency parsing is the word segment denoted as \(Seg = \{\text{token}_1, \ldots, \text{token}_m\}\), which contains \(m\) tokens. Therefore, a story can be represented as \(Y = \{Seg_1, \ldots\}\). For each story, we firstly identify the set of dependencies \(T = \{Seg_h, D_{\text{tag}}, Seg_i\}\), and then select target labels \(T_{\text{target}}\) to insert into the original stories. These target labels are \(\text{nsubj}\) (representing subjects), \(\text{root}\) (usually representing verbs), \(\text{dobj}\) (representing direct objects), and \(\text{pobj}\) (representing indirect objects following prepositions) (De Marneffe and Manning, 2008). The process is depicted as below:

\[
T_{\text{target}} = D_{\text{tag}} \in \{\text{nsubj}, \text{root}, \text{dobj}, \text{pobj}\} \\
\text{Tagger}(Seg_i) = \begin{cases} Seg_i, D_{\text{tag}} & \text{if } D_{\text{tag}} \in T_{\text{target}} \\
Seg_i & \text{otherwise} \end{cases} \\
Y^D = \text{Tagger}(Y, T_{\text{target}})
\]

where \(Y^D\) is a story with target dependency labels. For instance, the input “他们游历了所有的国家” ("They visited all the countries") will be tagged, and the output would be “他们<\text{nsubj}>游历<root>了所有的国家<\text{dobj}>" (They<nsubj> visited<root> all the countries<dobj>).

2.2 Semantic Denoising

To help neural networks understand the semantics of Chinese segments implicitly contained in sentences, we employ SimBERT (Su, 2020), which inputs a sentence, and outputs a similar sentence with the same meaning in order to generate a training corpus with large number of synonymous sentences. We therefore aim to train neural networks to resist the semantic noise introduced by different Chinese expressions. For instance, the compound words "去过" and "去了" both represent the meaning "went" in Chinese, in which “去” (go), with different auxiliary characters, may have the same meaning. As this phenomenon is ubiquitous in Chinese, we force the neural networks to denoise the changes in surface forms in order to better understand the semantics of these segments. Consequently, we obtain an augmented data corpus for semantic denoising:

\[
\{\ldots, Seg_i, \ldots\} = \text{SimBERT}(\{\ldots, Seg_i, \ldots\}) \\
\{Y^S_1, \ldots, Y^S_6\} = \text{SimBERT}(Y)
\]

where \(Seg_i\) is a synonym of \(Seg_i\). \(Y^S\) is a story that is different from \(Y\) but has the same input \(X\). We generate 6 similar stories for each \(X\), and train our neural generator on the enlarged corpus.
Table 1: Automatic evaluation of generated stories. The best score on each metric is highlighted in bold. \(w_i\) is the weight of each metric computed for the overall aggregate score. For all metrics, the higher the score, the better.

| Methods | Validation Set | Test Set |
|---------|----------------|----------|
|         | B-1  | B-2  | D-1  | D-2  | cover | order | Overall | B-1  | B-2  | D-1  | D-2  | cover | order | Overall  |
| ConvS2S | 29.23 | 10.38 | 3.45 | 21.79 | 14.81 | 25.34 | 16.08 | 29.00 | 10.14 | 1.60 | 13.95 | 15.45 | 25.77 | 15.19 |
| Fusion  | 29.22 | 10.34 | 3.39 | 22.67 | 17.41 | 26.55 | 16.5  | 28.77 | 10.22 | 1.47 | 14.12 | 17.10 | 26.36 | 15.40 |
| GPT2    | 30.43 | 14.87 | 10.95 | 44.38 | 60.90 | 55.52 | 28.43 | 30.17 | 14.91 | 7.62 | 36.87 | 60.87 | 55.90 | 27.62 |
|         | 30.17 | 14.91 | 7.62 | 36.87 | 60.87 | 55.90 | 27.62 |       |       |       |       |       |       |       |
| GPT2†   | 35.29 | 18.31 | 13.89 | 51.36 | 64.01 | 57.64 | 32.26 | 35.79 | 18.68 | 9.89 | 43.52 | 64.43 | 56.96 | 31.59 |
|         | 35.79 | 18.68 | 9.89 | 43.52 | 64.43 | 56.96 | 31.59 |       |       |       |       |       |       |       |
| PM      | 31.81 | 14.94 | 12.99 | 50.56 | 62.98 | 56.75 | 29.87 | 31.85 | 15.24 | 8.62 | 41.32 | 63.15 | 57.21 | 28.99 |
|         | 31.85 | 15.24 | 8.62 | 41.32 | 63.15 | 57.21 | 28.99 |       |       |       |       |       |       |       |
| PW      | 35.84 | 18.47 | 11.86 | 47.62 | 64.93 | 57.30 | 31.89 | 35.12 | 17.96 | 8.68 | 40.17 | 63.70 | 55.17 | 30.44 |
|         | 35.12 | 17.96 | 8.68 | 40.17 | 63.70 | 55.17 | 30.44 |       |       |       |       |       |       |       |
| Ours    | 44.40 | 25.49 | 17.35 | 62.47 | 88.93 | 64.72 | 41.41 | 44.82 | 25.88 | 12.31 | 53.21 | 89.15 | 67.05 | 40.78 |
|         | 44.82 | 25.88 | 12.31 | 53.21 | 89.15 | 67.05 | 40.78 |       |       |       |       |       |       |       |

\(F = \text{Encoder}(X)\) \hspace{1cm} \text{Tagger}(\{Y, Y^S_1, \ldots\}) \xrightarrow{\text{predict}} \text{Decoder}(F) \hspace{1cm} (6) (7)

where the maximum sequence length is set to 512 for both the \textit{Encoder} and \textit{Decoder}. LongLM is then fine-tuned with standard cross-entropy loss.

3 Experiment

3.1 Experiment Setup

\textbf{Dataset} We conduct our experiments on the OutGen task of LOT (Guan et al., 2022), a Chinese story benchmark which consists of 2427 high-quality filtered Chinese stories. Each input outline contains a sequence of 8 unordered phrases (i.e., their order does not necessarily reflect the order in which they would be present within a narrative). We follow the data split from the benchmark of 60/10/30 for training/validation/testing, respectively. The statistics are shown in Table 2.

\textbf{3.2 Baselines}

We compare our generation framework with a selection of competitive baselines, including the non-pretrained models ConvS2S (Gehring et al., 2017) and Fusion (Fan et al., 2018); pre-trained GPT2 models including GPT2\_base (Zhao et al., 2019a) and GPT2\_base (the latter of which is pretrained on the benchmark corpus) (Guan et al., 2022); PlotMachines (PM) (Rashkin et al., 2020b); Plan&Write (PW) (Yao et al., 2019); and mT5 (based on google/mt5-base) (Xue et al., 2021).

Specifically, the pre-trained models of baselines are implemented and restored from the prior works on the Chinese language. GPT2 based models are based on uer/gpt2-chinese-cluecorpussmall (Zhao et al., 2019b).

3.3 Implementation Details

We restore the publicly available checkpoint\(^4\) from Huggingface, and fine-tune LongLM\_base within our framework. LongLM has 12 attention heads and 12 hidden layers in each encoder and decoder, leading to a total of 223M parameters. We set the maximum sequence length to 512, the batch size to 3, and use a linear schedule to set the warm up step to 100 and the learning rate to 0.0001 for the Adam optimiser. All models are fine-tuned on 2 Nvidia RTX A5000 GPUs.

\[^4\]https://huggingface.co/thu-coai/LongLM-base
3.4 Evaluation Metrics

Following the LOT benchmark (Guan et al., 2022), we perform automatic evaluation on the metrics of BLEU-n ($B_n$) (Papineni et al., 2002), Distinct-n ($D_n$) (Li et al., 2015), Coverage ($cover$), and Order ($order$). The BLEU-n score measures the quality of generated text by comparing the degree of n-gram overlap with the ground-truth texts; the Distinct score measures the n-gram diversity of the generated text; Coverage is the same as ROUGE-L (Lin, 2004), which measures the recall rate between generated text and input outline phrases; and Order measures the difference between the positional orders of the input phrases in the generated text and the ground-truth text (which is calculated by dividing the number of positional order inversions in the generated story by the number of position pairs between any two phrases) (Guan et al., 2022). We compute the overall aggregate score with the metric weighting scheme presented in LOT.

3.5 Experimental Result

**Comparison with Baselines** As shown in Table 1, our proposed model substantially outperforms all competitive baselines by a considerable margin. We implement $\text{LongLM}_{\text{base}}$ (223M hyper-parameters) as our conditional generator. However, the results indicate our model can also significantly outperform $\text{LongLM}_{\text{large}}$ (1B hyper-parameters), on all metrics. Compared to the SOTA model ($\text{LongLM}_{\text{large}}$), our proposed model achieves up to a 10% improvement on several metrics for both the validation and test sets, and around 5% for the overall score. Additionally, when compared to $\text{LongLM}_{\text{base}}$, our model demonstrates a performance uplift of around 10% on the overall score.

The performance improvements seen on BLEU-n and Coverage indicate that our generated stories have a higher degree of overlap with the reference stories. Considering the input outline is unordered, this indicates that via the awareness of dependencies and semantics, our proposed model can better leverage syntactic features, and generate more fluent narratives as a result. The scores on Order (computed by the order of outlines in the generated stories compared to the reference), further demonstrate the improvement on language discourse. Meanwhile, the diversity of stories is also substantially raised, for which we argue that semantic denoising contributes significantly.

Considering the results as a whole, the significant improvements of our model over existing baselines demonstrates that the enhanced capturing of dependencies and semantics contribute to the language understanding task. This is particularly apparent for Chinese, where expressions are more ambiguous due to the lack of explicit delimiters. Using this increased level of understanding, conditional generators can therefore generate more fluent and diverse stories.

**Ablation Study** We conduct ablation experiments presented in Table 3 to analyse the individual contributions of each module. We observe that the enhanced feature capturing of both the dependencies and semantics substantially improves on the original neural generator, and combining both approaches further improves performance. This indicates that these two features largely perform different functions that contribute to language generation. Whilst our proposed model outperforms all ablated models when considering most metrics, performance of a single module on some metrics is still close to or even slightly better than the combined model (e.g., on coverage). This phenomenon implies that the two proposed modules may have a small degree of shared function when exploiting features from text. In addition, insufficient training may also lead to the inadequacy of incorporating both features for decoding. We leave further study of incorporating both features to future work.

**Case Study** Several generated Chinese stories are presented in subsection A.1 to further demonstrate the effectiveness of our framework.

### Table 3: Automatic evaluation for the ablation study. **Dependencies** denotes the Dependency Tagging module, and **Semantics** denotes the Semantic Denoising module.
4 Conclusion

We propose a novel story generation framework for Chinese, which includes a dependency tagging module, a semantic denoising module, and a neural conditional generator. We aim to improve the generation of Chinese through more effectively incorporating the features of dependencies and semantics. The performance improvements shown in our experiments and ablation study demonstrate that these features significantly benefit the task of Chinese story generation.

Acknowledgements

Chen Tang is supported by the China Scholarship Council (CSC) for his doctoral study (File No.202006120039). Tyler Loakman is supported by the Centre for Doctoral Training in Speech and Language Technologies (SLT) and their Applications funded by UK Research and Innovation [grant number EP/S023062/1]. We also gratefully acknowledge the anonymous reviewers for their insightful comments.

References

Faeze Brahman and Snigdha Chaturvedi. 2020. Modelling protagonist emotions for emotion-aware storytelling. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 5277–5294, Online. Association for Computational Linguistics.

Guanyi Chen, Kees van Deemter, and Chenghua Lin. 2018. SimpleNLG-ZH: a linguistic realisation engine for Mandarin. In Proceedings of the 11th International Conference on Natural Language Generation, pages 57–66.

Marie-Catherine De Marneffe and Christopher D Manning. 2008. The stanford typed dependencies representation. In Coling 2008: proceedings of the workshop on cross-framework and cross-domain parser evaluation, pages 1–8.

Angela Fan, Mike Lewis, and Yann Dauphin. 2018. Hierarchical neural story generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 889–898.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. In International Conference on Machine Learning, pages 1243–1252. PMLR.

Seraphina Goldfarb-Tarrant, Tuhin Chakrabarty, Ralph Weischedel, and Nanyun Peng. 2020. Content planning for neural story generation with aristotelian rescoring. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4319–4338, Online. Association for Computational Linguistics.

Jian Guan, Zhuoer Feng, Yamei Chen, Ruijun He, Xiaoxi Mao, Changjie Fan, and Minlie Huang. 2022. LOT: A story-centric benchmark for evaluating Chinese long text understanding and generation. Transactions of the Association for Computational Linguistics, 10:434–451.

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

Han He and Jinho D. Choi. 2021. The stem cell hypothesis: Dilemma behind multi-task learning with transformer encoders. In Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 5555–5577, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2015. A diversity-promoting objective function for neural conversation models. CoRR, abs/1510.03055.

Yucheng Li, Chenghua Lin, and Frank Guerin. 2022. Cm-gen: A neural framework for chinese metaphor generation with explicit context modelling. In Proceedings of the 29th International Conference on Computational Linguistics.

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

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In Proceedings of the 40th annual meeting of the Association for Computational Linguistics, pages 311–318.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020a. PlotMachines: Outline-conditioned generation with dynamic plot state tracking. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4274–4295, Online. Association for Computational Linguistics.

Hannah Rashkin, Asli Celikyilmaz, Yejin Choi, and Jianfeng Gao. 2020b. Plotmachines: Outline-conditioned generation with dynamic plot state tracking. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 4274–4295.
A Appendix

A.1 Case Study

In Table 4 we present an example for the basis of a case study. Table 5 presents the generated stories from the neural generation models, including the SOTA baseline LongLM_{base}, our proposed framework, and its ablated models.

Firstly, with large-scale pre-training on narrative corpora, the generated stories have relatively less repetition and diversity problems than traditional text generation methods. The main issues are now located in linguistic aspects such as fluency, coherence, and relevance. It can be observed that the generated story from the SOTA baseline model suffers from the ambiguity of the Chinese language, which leads to grammatical and semantic errors. For instance, the sentence “从前，有个挑水夫，他把路旁的半桶水送到主人家的破桶留意路旁” (Once upon a time, there was a water-carrier who sent half a bucket of water sprinkled by the roadside to the broken bucket at the master’s house) has grammatical errors. This may result from inadequate understanding of the dependency roles of each part of the sentences, which leads to misusing two verb phrases (“sent”, “keep and eye on”). For the same reason, the linguistic ambiguity makes the model struggle to capture the semantic meaning of each sentence constituent. For example, the sentence “结果，路旁就完好无损了” (As a result, the roadside was intact) contains no grammatical errors, but also makes no sense to the story. It can be intuitively supposed that the key words “the roadside” and “intact” in the given outline are directly composed here by the neural model without understanding their semantics.

Regarding the ablation study, similar errors can also be observed in each ablated model: (1) “主人让他把破桶留意路旁撒的水 送去主人家” (The master asked him to send the water the broken bucket noticed sprinkled by the roadside to the master’s house.), in the story generated by - w/ Dependencies, also has obvious grammatical errors; (2) - w/ Semantics generates the sentence “主人对他道歉，并把那水送 到了路旁撒了一次又一次网。” (The master apologised to him and sent the bucket of water to the roadside for casting the net again and again.), which is free of grammatical errors, but does not make sense semantically.

Furthermore, when comparing sentences containing the same key words from outlines in dif-
Outline: "破桶留意路旁", "只能剩下半桶水", "水送到主人家", "挑水夫道歉", "路旁撒", "挑水夫说", "趟挑运", "完好无损"

"the broken bucket keeps an eye on the roadside", "only half a bucket of water is left", "deliver water to the master’s house", "the water-bearer apologises", "sprinkled by the roadside", "the water-bearer said", "travel to pick up", "intact"

Reference Story: 挑水夫有两个水桶，一个桶有裂缝，另一个完好无损。每趟挑运之后，好桶总是能将满满一大桶水送到主人家中，但是破桶却只能剩下半桶水。破桶非常羞愧，一天，它对挑水夫道歉。挑水夫并没有生气，他让破桶留意路旁的花朵。他们走在山坡上，破桶看到缤纷的花朵，开满在路的一旁。挑水夫说，只有破桶的那一边有花，好桶的那一边却没有。原来挑水夫知道破桶的缺陷，因此善加利用。在破桶那边的路旁撒了花种，每隔三天浇一次，破桶就替它一路浇了花。如果不是因为破桶，主人的桌上也没有那么好看的花朵了。

The water-bearer had two buckets. One bucket is broken and the another is intact. After each pick-up, the good bucket can always deliver a full bucket of water to the master’s house, but the broken bucket only has half a bucket of water left. The broken bucket feels very ashamed. One day, it apologised to the water bearer. The water-bearer was not angry, he told the broken bucket to keep an eye on the flowers by the roadside. As they walked down the hillside, Broken bucket saw colorful flowers that filled the side of the road. The water-bearer said that there were flowers only on the side of the broken bucket, but not on the side of the good bucket. It turned out that the water-bearer knew about the defects of the broken bucket, so he made good use of it. Water-bearer sowed flower seeds on the roadside over the broken bucket. Every time he came from the stream, the broken bucket would water the flowers along the way. If it weren’t for the broken bucket, there would not be such beautiful flowers on the master’s table.

Table 4: An example of an outline and its reference story.

different generated stories, it can be observed that our proposed techniques for dependency parsing and semantic denoising indeed improve the quality of generated sentences. For example, for the key word “水送到主人家” (deliver water to the master’s house), the aforementioned LongLM and - w/ Dependencies all generate a sentence with grammatical errors. However, the sentences “挑水人问他为什么不把水送到主人家” (The water-bearer asked him why he didn’t deliver the water to the master’s house.) by - w/ Semantics and “挑水的人觉得，这趟挑运真是完好无损” (Water-bearer felt that the trip was really intact.) by Ours, are all free of grammatical errors.

Compared to other generated stories, the story from our proposed model contains less errors, and reads more fluently and coherently. Therefore, with the analysis of these errors, we can confirm that current neural models are likely to suffer from the challenge of parsing dependencies and semantics in Chinese narratives. However, our proposed generative framework can improve this uncertainty brought by language ambiguity, and consequently generates a better story.
Once upon a time, there was a water-bearer who wanted to deliver half a bucket of water sprinkled to the master’s house. The water-bearer asked him why he didn’t deliver the water to the master’s house. The master said, “travel to pick up”, “intact” water to the master’s house”, “The water-bearer apologises”, “Sprinkled by the roadside”, “the water-bearer said”, “travel to pick up”, “intact”

Once upon a time, there was a water-bearer who wanted to deliver half a bucket of water sprinkled to the master’s house. The water-bearer asked him why he didn’t deliver the water to the master’s house. The master said, “travel to pick up”, “intact” water to the master’s house”, “The water-bearer apologises”, “Sprinkled by the roadside”, “the water-bearer said”, “travel to pick up”, “intact”

Outline:
"the broken bucket keeps an eye on the roadside", "Only half a bucket of water is left", "deliver water to the master’s house", "The water-bearer apologises", "Sprinkled by the roadside", "the water-bearer said", "travel to pick up", "intact"

Long LM: From now on, there was a water-bearer who went half a bucket of water sprinkled by the roadside to the master’s house and the broken bucket kept an eye on the roadside. The master was very angry, so he sent the water to the side of the road, and also sent the water up. As a result, the roadside was intact. This time, the water-bearer boss was not happy. He apologised to his master and said that he had done nothing good. When the master heard this, the master felt that the matter was a mistake and only half a bucket of water could be left. Therefore, he decided to attribute all the harvest of this fortune to the master. From then on, the master never dared to bully him again, never bullied him again.

-w Dependencies From now on, there was a water-bearer who went half a bucket of water sprinkled by the roadside to the master’s house. The master asked him to send the water the broken bucket noticed sprinkled by the roadside to the master’s house. So, the master sprinkled water on the roadside for more than ten minutes, until the water was delivered to the master’s yard. On this day, the master asked them to carry water together again. They came back from a single trip, and all trips were all intact, and it was like this for more than ten days. Finally, they went to pick up the dozen or so buckets of water that were brought back. Now, the two of them can go to the roadside to get a bucket of water. Since then, the two of them have never had more than half a bucket of water, and have never encountered other people who came to collect water.

-w/ Semantics There was a water-bearer who had only half a bucket of water left because it was intact. The water-bearer asked him why he didn’t deliver the water to the master’s house. The master said that this trip took only three seconds to deliver the water. Not long after, the water-bearer starved to death! The master apologised to him and sent the bucket of water to the roadside for casting the net again and again. In the end, the man was saved, and he finally apologised to the parents of the master.

Ours From now on, there was a water-bearer who wanted to deliver half a bucket of water sprinkled by the roadside to the master’s house. The water-bearer felt that the trip was really intact. So he went to the roadside to pick up a broken bucket, hoping to use it to deliver the water. As a result, the water in the broken bucket was sprinkled all over. When the master found out, he was very angry. Then, the master ordered the water-bearer to be drowned by the roadside.

Table 5: The given outline and corresponding generated stories for the case study.