Creative storytelling with KGs

Xinran Yang

Vrije Universiteit Amsterdam, 1081 HV Amsterdam, Netherlands
x6.yang@student.vu.nl

Abstract. Automated story generation is a popular and well-recognized task in the field of natural language processing. Although the existing language models show the great capability of generating coherent text, the output is usually monotonous and easily made a digression. In this study, we try to combine knowledge graphs with the state-of-the-art language model, OpenAI GPT-2, to build a creative story generation system named DICE, which uses external knowledge graphs to provide context clues and implicit knowledge to generate more reasonable and creative stories. The DICE system extracts keywords from the input prompt, then extends keywords by using external knowledge graphs, and finally feeds the enriched keyword sets to the language model and generates stories. The evaluation for the quality of generated stories involves both human feedback and automated linguistic analysis tools.

Keywords: Story generation · Knowledge graph · Language model · Natural language generation.

1 Introduction

1.1 Motivation

After Large-scale pre-trained language modes like OpenAI GPT-2 (Radford et al. 2019) and BERT (Devlin et al. 2018) being released in recent years, it has become possible for machines to generate a paragraph of coherent and creative text according to a given topic. Before that, Natural Language Generation (NLG) is only considered to be used in generating relative text based on the existing templates, but not for creative works like writing fiction and music. In November 2019, OpenAI officially released the full version of GPT-2 with 1.5 billion parameters[1], and the language model can generate coherent paragraphs of text on the topics chosen by the user. Meanwhile, more and more people get interested in applying Knowledge Graphs (KGs) to language models because KGs can not only provide factual knowledge to automatically generated texts but also improve the performance of the language models (Logan et al. 2019). Since a pre-trained language model has the "innate talking skill" and can be employed as a generator (Chen et al. 2019), and meanwhile, KGs can provide contents and structures for story generation, so in this research, we are going to take

[1] https://openai.com/blog/gpt-2-1-5b-release

advantage of both technologies. A study is performed on how to automatically generate a story using KGs based on existing language models. The study focuses on the specific role of KGs when generating stories and how to use KGs to improve the performance of automated story generation.

1.2 Problem Definition

Natural language processing (NLP) can be classified into two fields: Natural language understanding (NLU) and NLG. According to types of input, we can also divide the current popular technologies of NLG into three categories: Text2Text NLG (the input is text), Kb2Text NLG (the input is knowledge base structured data), and Mm2Text NLG (the input is image, and Mm is short for Multi-modal) (Amidei et al. 2018). The automatic story generation process can be regarded as Kb2Text NLG or neural data-to-text generation (Logan et al. 2019). Meanwhile, the task requires not only NLG but also NLU since understanding context clues and handling implicit knowledge also play an important role when automatically generate a reasonable story (Guan et al. 2019). However, current NLG systems applied by industrial more focus on the "Generation" instead of "Creation". For example, automated report generation uses the templates and slot-filling rules to automatically generate reports based on the real-time data (Chen et al. 2019). Although state-of-the-art language models such as BERT and GPT-2, show the great capability of generating coherent text based on a topic with near-perfect grammatical syntax and punctuation, the generated text can hardly find related but implicit topics to extend the content (Guan et al. 2019). This research aims to analyze how to use the current popular language models like OpenAI GPT-2 in conjunction with KGs to generate creative short stories. We try to find a practical way of the implementation, analyze the special role of KG during the generation, and evaluate the effect of applying KGs to story generation.

1.3 Research Questions

RQ1: How to automatically generate a story using knowledge graphs?
Sub-question of RQ1:
SQ1: How to combine existing language models (especially GPT-2) with KGs?

RQ2: What are the advantages and disadvantages of using knowledge graphs to automatically generate a story?
Sub-question of RQ2:
SQ2: What kind of information can be provided by knowledge graph and be actually used by language model during story generation?

To address RQ1, I considered how to combine existing popular language models (especially GPT-2) with KGs to generate a story. With respect to RQ2, I considered the effects of applying KGs to automatic creative storytelling and what kind of information or functions can KGs contribute to the automatic writing process.

1.4 Scientific Contribution

Although there were a few previous studies trying to use language models in conjunction with KGs, very few studies tried to combine them for creative storytelling, such as generating short novels or fictions. Since KGs can store information about relations between entities, it is very useful to use KGs to generate novel-like text especially when there are complex relationships between characters. Moreover, this research especially focuses on combining state-of-the-art Transformer architecture, OpenAI GPT-2, with KGs. Because the GPT-2 models are very new and the full version of GPT-2 was just released in November 2019, so there is no study performing the combination before. This study will find a practical way to take advantage of both technologies. We believe this study can give a clear picture of how to use external KGs to generate coherent and creative short stories.

2 Related Work

Story generation is a knowledge-intensive process (Li et al. 2013). In particular, open story generation requires artificial intelligence systems to create narratives about any topic without a pre-defined domain model (Li et al. 2013). Meanwhile, a creative story should be both novel and appropriate (Sternberg 1999). While for existing NLG systems, they are mostly weak AI systems and they are often limited when the tasks requiring higher levels of creativity and originality (Jain et al. 2017). Pre-trained language models based on Large Transformer Architectures (Vaswani et al., 2017), such as GPT-2 and BERT, maybe a potential solution for this problem. These language models show impressive text generation capabilities that can achieve state-of-the-art results without extra training (Keskar et al. 2019). However, these language models perform poorly when capturing the long tail of rare entities such as numbers and dates (Logan et al. 2019). Moreover, these models are unable to build context clues and use implicit knowledge to generate a reasonable story ending (Guan et al. 2019).

This situation can be improved by combining language models with KGs, where the former can facilitate the knowledge extracted from KGs. For example, the Knowledge Graph Language Model (KGLM) is a natural language model with a KG to access the external source of facts (Logan et al. 2019). Ostendorff et al. (2019) enriched BERT with KG embeddings for document classification and got better results than the standard BERT approach. Liu et al. (2019) proposed a knowledge-enabled language representation model (K-BERT) using KGs to provide domain-specific knowledge. Meanwhile, Koncel-Kedziorski et al. (2019) introduced a novel graph transforming encoder for graph-to-text generation. This research is also inspired by the work of Hsu et al. (2019) who proposed the distill-enrich-generate framework, and this three-stage framework allows Transformer architectures to take advantage of external KGs through all the three phases to generate stories. However, the studies above all focus on using KG as an external knowledge base to make the generated text more complete instead of using it as a source of inspiration to provide main topics during the story generation. This study will focus on how to use KG to provide relative and interesting topics for language model to generate a creative story.

3 Research Methods

A complete story should at least contain the following elements: characters, places, time, objects, and actions. Although it is not always the case to strictly follow 5W + 1H (who, where, when, what, why, how) model (Mosannenzadeh et al. 2017) to make a clear storytelling, the above five elements are essential in narrative. Like the dice storytelling game, we can use 5 dices to represent the five elements. Then we can roll all the 5 dices and generate 5 elements from each aspect, and finally, use all these elements to invent a story. KGs can provide not only external information of these elements but also ontological characteristics of each element. So the basic idea in this research is using KGs to enrich the keyword sets given by the input prompts and generate a script, and then we will use the script to feed the language model to generate a complete and coherent story. Since the idea comes from the dice storytelling game, we name this story generation system as DICE. The DICE system contains two layers: the knowledge graph layer and the language model layer. The knowledge graph layer can be used to provide relative and creative topics while the language model layer can be used as a generator for story generation. Figure 1 shows the workflow of the DICE system.

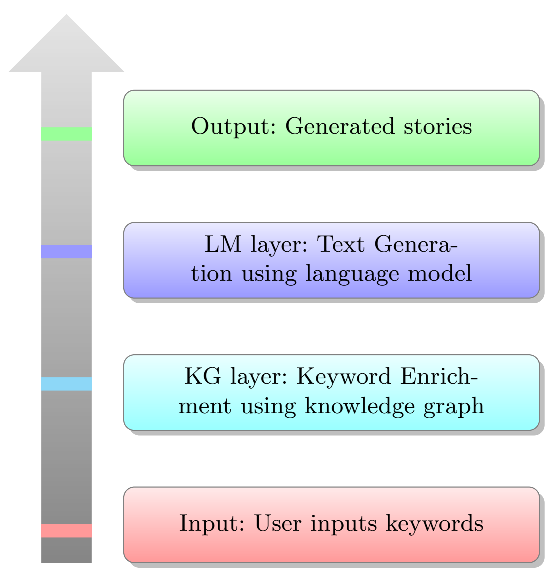

Fig. 1. Workflow of the DICE System

3.1 Keyword Enrichment: Creativity from Knowledge Graphs

More keywords can lead to more abundant content of the generated text. Former studies indicate that information from a commonsense knowledge base is very helpful for many tasks in NLG, such as dialogue generation (Liu et al. 2018) and story completion (Chen J et al. 2019). This is because many hidden relationships among keywords can be uncovered by using KGs (Chen J et al. 2019). On the other hand, KGs should provide not only commonsense from the real world, but also commonsense from the virtual story world. For example, we can use KGs to describe the relationship among characters in the story, or we can use it to add personality to each role. Applying this information can make the generated story more coherent and more reasonable.

Sometimes, the creativity of the story can be too vague to be measured. However, creativity can be seen as a matter of probability: the more combination of different ideas (despite the quality of the ideas), the more creative stories are likely to be generated. In this case, KGs can provide us countless related ideas or topics to generate a story, which means each "dice" has thousands of faces instead of 6. In this research, we use ConceptNet[2] (Speer et al. 2017) as an external knowledge graph resource to capture the relevant knowledge and enrich keywords.

3.2 Text Generation: Language Model as a Generator

We use the Transformer architecture, OpenAI GPT-2, to transform the keyword sets from the last step into stories. The Transformer is a deep machine learning model and supports much more parallelization than recurrent neural networks (RNNs) during training (Vaswani et al. 2017). Experimental results show Transformer architectures, such as BERT and GPT have the efficacy on various NLP tasks (Wang et al. 2019). The only difference between GPT (or GPT-2) and BERT is that BERT is bidirectional while GPT uses masked self-attention heads (Wang et al. 2019). The reason why choosing GPT-2 over BERT is that GPT-2 has better open-source support, tools like gpt-2-simple[3] can be easily used to fine-tune the models and generate scripts.

Since the pre-trained GPT-2 models were trained mostly in the English context, so in this research, we use English as the main language to generate stories. Meanwhile, we choose the 124M model (the default) of GPT-2 since the limitation of GPUs, but this model can also provide a good balance of speed and creativity during the text generation.

3.3 Evaluation

The generated stories from the DICE system will be compared with human-authored stories (gold standard) and stories generated by GPT-2 (baseline). To evaluate the quality of the generated stories, we will combine human feedback with automated evaluation.

[2] http://conceptnet.io/
[3] https://github.com/minimaxir/gpt-2-simple

DICE vs. Human A keyword set will be provided to both a human and the DICE system to create a story, then we compare the results.

DICE vs. GPT-2 We construct one or two sentences containing all the entities in the keyword set, and we use these sentences as an input for GPT-2 to generate a story. Then we use the same keyword set to generate a story using DICE and compare the results.

Evaluation Metrics The evaluation will focus on two aspects of the generated output: story-independent metrics and story-dependent metrics (Roemmele et al. 2017). Story-independent metrics, including grammaticality and syntactic complexity, will be used to analyze the quality of the generated output without considering its context; whereas story-dependent metrics, including lexical cohesion, entity coreference, and creativity, will be used to evaluate the generated stories with reference to the context (Roemmele et al. 2017). Some automated linguistic analysis tools such as LanguageTool[4] will also be used to measure the quality of the generation.

References

1. Amidei, J., Piwek, P., & Willis, A. (2018). Evaluation methodologies in Automatic Question Generation 2013-2018.
2. Chen, J., Chen, J., & Yu, Z. (2019, July). Incorporating structured commonsense knowledge in story completion. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 6244-6251).
3. Chen, Z., Eavani, H., Liu, Y., & Wang, W. Y. (2019). Few-shot NLG with Pre-trained Language Model. arXiv preprint arXiv:1904.09521.
4. Devlin, J., Chang, M. W., Lee, K., & Toutanova, K. (2018). Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805.
5. Guan, J., Wang, Y., & Huang, M. (2019, July). Story ending generation with incremental encoding and commonsense knowledge. In Proceedings of the AAAI Conference on Artificial Intelligence (Vol. 33, pp. 6473-6480).
6. Hsu, C. C., Chen, Z. Y., Hsu, C. Y., Li, C. C., Lin, T. Y., Huang, T. H. K., & Ku, L. W. (2019). Knowledge-Enriched Visual Storytelling. arXiv preprint arXiv:1912.01496.
7. Jain, P., Agrawal, P., Mishra, A., Sukhwani, M., Laha, A., & Sankaranarayanan, K. (2017). Story generation from sequence of independent short descriptions. arXiv preprint arXiv:1707.05501.
8. Keskar, N. S., McCann, B., Varshney, L. R., Xiong, C., & Socher, R. (2019). Ctrl: A conditional transformer language model for controllable generation. arXiv preprint arXiv:1909.05858.
9. Koncel-Kedziorski, R., Bekal, D., Luan, Y., Lapata, M., & Hajishirzi, H. (2019). Text Generation from Knowledge Graphs with Graph Transformers. arXiv preprint arXiv:1904.02342.

[4] https://github.com/languagetool-org/languagetool

10. Li, B., Lee-Urban, S., Johnston, G., & Riedl, M. (2013, June). Story generation with crowdsourced plot graphs. In Twenty-Seventh AAAI Conference on Artificial Intelligence.
11. Liu, S., Chen, H., Ren, Z., Feng, Y., Liu, Q., & Yin, D. (2018, July). Knowledge diffusion for neural dialogue generation. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers) (pp. 1489-1498).
12. Liu, W., Zhou, P., Zhao, Z., Wang, Z., Ju, Q., Deng, H., & Wang, P. (2019). K-bert: Enabling language representation with knowledge graph. arXiv preprint arXiv:1909.07606.
13. Logan, R., Liu, N. F., Peters, M. E., Gardner, M., & Singh, S. (2019, July). Barack's wife hillary: Using knowledge graphs for fact-aware language modeling. In Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics (pp. 5962-5971).
14. Mosannenzadeh, F., Bisello, A., Vaccaro, R., D'Alonzo, V., Hunter, G. W., & Vettorato, D. (2017). Smart energy city development: A story told by urban planners. Cities, 64, 54-65.
15. Ostendorff, M., Bourgonje, P., Berger, M., Moreno-Schneider, J., Rehm, G., & Gipp, B. (2019). Enriching BERT with Knowledge Graph Embeddings for Document Classification. arXiv preprint arXiv:1909.08402.
16. Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., & Sutskever, I. (2019). Language models are unsupervised multitask learners. OpenAI Blog, 1(8).
17. Roemmele, M., Gordon, A. S., & Swanson, R. (2017, August). Evaluating story generation systems using automated linguistic analyses. In SIGKDD 2017 Workshop on Machine Learning for Creativity (pp. 13-17).
18. Speer, R., Chin, J., & Havasi, C. (2017, February). Conceptnet 5.5: An open multilingual graph of general knowledge. In Thirty-First AAAI Conference on Artificial Intelligence.
19. Sternberg, R. J. (Ed.). (1999). Handbook of creativity. Cambridge University Press.
20. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. In Advances in neural information processing systems (pp. 5998-6008).
21. Wang, C., Li, M., & Smola, A. J. (2019). Language models with transformers. *arXiv preprint arXiv:1904.09408*.

A Project Plan

Master Project

Project Schedule

Start Week	Feb 3, 2020

Week	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	Notes
Starting	Feb 3	Feb 10	Feb 17	Feb 24	Mar 2	Mar 9	Mar 16	Mar 23	Mar 30	Apr 6	Apr 13	Apr 20	Apr 27	May 4	May 11	May 18	May 25	Jun 1	Jun 8	Jun 15	
Grasp basic techniques for knowledge-based NLP																					Tutorial from http://hybridnlp.expertsystemlab.com/tutorial/
Explore GPT and build a command line tool to generate coherent text using GPT-2																					
Develop feature of keyword extraction																					
Develop feature of keyword enrichment using KGs																					
Combine the KG layer with the command line tool																					
Fix bugs and fine-tune the system																					
Prepare stories (respectively generated by DICE, GPT-2, and written by human)																					
Make a set of criteria and evaluate stories																					
Write the research paper																					