SI630 Project Update

Fangzhe Li
University of Michigan / 536 S. Forest Ave
School of Information / Ann Arbor, MI
fangzhel@umich.edu

1 Introduction

Classical Chinese poems excel themselves by their conciseness and elegance. It is always an interesting task to create poem. In homework 1, we have done a binary-classification problem in English. In this project, I initiate a task of creating Classical Chinese poems from several keywords. Through this project, I hope we can see the difficulties and challenges of doing nature language processing on a different language as mentioned in the first lecture of our class. Except for the reason that we want our "poet" to surprise us with beautiful rhythms, the other natural language generation tasks can also be benefited from the research on poem generation. (**?**) The problem to solve is that given some keywords, the output is supposed to be a quatrain with 7 Chinese characters in each line which focuses on the topic of keywords and fits the rhythm.

2 Task Definition and Data

2.1 Task Definition

I got some collections of classical Chinese poem as the corpus. I want to get a model that given some keywords as the first few characters of the poem from users as input and outputs a quatrain with either 5 or 7 Chinese characters in each line which focuses on the topic of keywords and fits the rhythm. Let $X = (X_1, X_2, \ldots, X_n)$ be a sequence of keywords, where X_j represents jth word in X, the output will be $Y = (Y_1, Y_2, \ldots, Y_{28})$, where y_j represents jth character in Y.

2.2 Data

I found the data on the public repo in GitHub: https://github.com/chinese-poetry/chinese-poetry/tree/master/json There are 315,000 poems stored in json. Each data contains id, author, paragraphs and title.

3 Methodology

3.1 Preprocess

Firstly, since the data in the dataset is all traditional Chinese. I used OpenCC: https://github.com/BYVoid/OpenCC as the tool to make this convertion. Also, since we focuses on the generation of 4-line poems, we cut all the 8-line poems into two 4-line poems, which is technically right because in the context of classical Chinese poem, 4-line poems are called "half of 8-line poems". For those poems which don't below to either 4-line poems and 8-line poems, they are collected as the source of chinese vector embedding fine-tune since even though these poem has different format, their word style fit the way of 4-lines poem very much. Then we can get around 280,000 "4-line poems". To make them align with each other, we add ¡start¿ and ¡EOF¿ at the beginning and end of the text. Also, for those poems who has 5 characters in a line, 8 blank elements are added right before ¡start¿. So now every input will be 34 characters. Here only paragraphs in the dataset is used. More information like authors and titles can be used in the improved version later.

3.2 Model

I plan to use a pre-trained Chinese embedding and fine-tune it with the extra data I have from last section and then use LSTM as the main part of model, the input dimension will be the length of embedding size, the hidden dimension I will start by 256, the layer number will be 2 as the start. Finally, I will have a linear layer map from hidden dimension back to embedding size.

4 Related Work

Generally, according to the methodology, works can be divided into three main categories: (1)

templated-based method (2) statistical machine translation model (He et al., 2012) (3) deep learning method(Li et al., 2018) and (Zhang et al., 2017). In (He et al., 2012), the authors use the keywords as the input and generate the poem sentence by sentence through a phrase-based SMT model. In this way, each sentence takes all the previous sentences into consideration to ensure coherence between lines. Also, the authors make a comparison between BLEU and human evaluation to show BLEU metric is a good way to evaluate poem generation models. In (Li et al., 2018), the authors use CAVE to generate novelty and discriminator to ensure coherence. They combines CAVE with adversarial training. In (Zhang et al., 2017), a memory-augmented neural model is used to solve the problem that the model only generate poems based on general rule and has very few innovations. In (Ghazvininejad et al., 2016), the authors use an interesting preprocess before using encoder-decoder model, they asked some input as the keywords of the poem, they make full use of word2vec to find words related to keywords and choose the words fit the rhyme. Put these words to the encoder and let decoder generate the poem. I think I can try this kind of preprocess to fix the output in some spot to make a better performance on rhyme. In (Yan, 2016), the author uses a line-by-line generation encoder-decoder system, the interesting part in this paper is that the author tries to mimic the process of human polishing the poem by use the input of the first round (from writing intention) and the output poems from the the first round as the input of next round to mimic polishing. For LSTM model, I am not quite sure if I can do the similar process to mimic polishing but it is an interesting thought.

5 Evaluation

Since it is very subjective to judge whether a poem is a good way, it is very hard to find a good criteria to assess the poems generated. After reading the literature in the related works part, we find there are two main ways to evaluate the machine-generated poems: human expert evaluation and BLEU (Papineni et al., 2002), which is a metric used widely in machine translation.

Human expert evaluation: In most of paper related to poem generation, human expert evaluation is used as the main criteria for evaluation because the poem is relatively complicated form of literature and it is very hard to quantitatively evaluate. Human evaluation can be done by setting several group of standard: Conciseness, Elegance, Rhythm. Another task can be done by human evaluation is that let experts tell if one poem is written by human poets or generated by our model, we can get the accuracy as the criteria. I expect to have the accuracy of distinguishing below 70

	Baseline
Number of poems tested	10
Mistakes of human	0
Mistake rate	0%

BLEU: Another quantitative way to evaluate poems is the BLEU. BLEU is initially used in machine translation to evaluate the similarity between human translation and generated machine translation. Here a slightly different way will be used: first keywords is extracted from a human-written poem, then these keywords are put into model as input, we compare the BLEU between the output the the original poem. Here a low BLEU value is expected because there are more ways to write a poem compared to general translation. So BLEU is better used to compare the result of different poem generation model. In (He et al., 2012), the author made an analysis to show BLEU actually a good criteria because it shows some relationship with the result of human evaluation.

Baseline As is discussed above, BLEU will only make sense when you have another system to compare with. So for the future evaluation, I will try to find some code developed by other people to compare my result with theirs in their paper. Another important baseline is that random select 5 poems from the dataset, since it is a huge dataset, it is very unlikely these poems are famous ones that already learnt by people by heart. Using the first line as the input, use this model to generate another five poems and mix with the 5 human-written poems, let people decide which ones belong to human and which ones belong to the model. For this task, given the first line and generate a 4-line poem, the baseline is simply repeating the first line for the rest of 3-line. Using the human evaluation method, the following result is got.

6 Work Plan

I had finished the data preprocess part and now looking for the good Chinese embedding. From

week 9 to 12, I am expected to finish the implementation of the model. For week 9, I should finishing fine-tuned embedding. For week 10, model implementation should be done. For week 11, I should finish the first training and start to improve it.From week 13 to 15, I will finish the presentation and evaluation part as well as the report for project.For week 13, I should finalize my project and start evaluating it. For week 14, I should finish all the evaluation. For week 15, I should finish my report.

References

Marjan Ghazvininejad, Xing Shi, Yejin Choi, and Kevin Knight. 2016. Generating topical poetry. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1183–1191, Austin, Texas. Association for Computational Linguistics.

Jing He, Ming Zhou, and Long Jiang. 2012. Generating chinese classical poems with statistical machine translation models. In *Proceedings of the Twenty-Sixth AAAI Conference on Artificial Intelligence*, AAAI'12, page 1650–1656. AAAI Press.

Juntao Li, Yan Song, Haisong Zhang, Dongmin Chen, Shuming Shi, Dongyan Zhao, and Rui Yan. 2018. Generating classical chinese poems via conditional variational autoencoder and adversarial training. pages 3890–3900.

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 on Association for Computational Linguistics*, ACL '02, page 311–318, USA. Association for Computational Linguistics.

Rui Yan. 2016. I, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*, IJCAI'16, page 2238–2244. AAAI Press.

Jiyuan Zhang, Yang Feng, Dong Wang, Yang Wang, Andrew Abel, Shiyue Zhang, and Andi Zhang. 2017. Flexible and creative Chinese poetry generation using neural memory. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1364–1373, Vancouver, Canada. Association for Computational Linguistics.