Learning to be Poetic: Automatic Generation of Chinese Song Ci Using RNN

Nan Du
Michigan State University
East Lansing, MI 48823, USA
dunan@msu.edu

Wei Wang
Michigan State University
East Lansing, MI 48823, USA
wangwe90@msu.edu

Zhuangdi Zhu
Michigan State University
East Lansing, MI 48823, USA
zhuzhuan@msu.edu

ABSTRACT

Judy: To be re-written In this project, we are going to develop a poem generator for Ci poetry. A system based on recurrent neural network (RNN) will be implemented to solve the sequence-to-sequence learning problem. We will also compare the performance of this model with the traditional automatic method that used to generate poem. We hope that our system can learn the complete rule from training dataset without any given constraints and generate poems with meaningful syntax that following the special rules for rhyme and tone in the Ci.

Keywords

Song Ci (poetry); Recurrent Neural Network; sequence-to-sequence learning problem

1. PROBLEM DESCRIPTION

1.1 Motivation

Judy: To be extened In this project, we propose and evaluate different approaches to automatically generate Chinese poems. Especially, we study how to automatically generate Chinese Ci using machine learning skills. Ci are one of the most important genres of Chinese classical poetry. As a precious cultural heritage, not many of them have been passed down onto the current generation. Therefore, the study of automatic generation of Ci is meaningful, not only because it supplements entertainment and education resources to modern society, but also because it demonstrates the feasibility of applying artificial intelligence in Art generation.

1.2 Background

Judy: To be extened Ci is a form of Chinese classical poetry. It arose with the so-called banquet music in Tang dynasty and reached its peak one hundred years later, as a major alternative to Shi poetry [2] .

Derived from the structure used in Tang poetry, Ci follows strict rule determining the number of characters for different lines, the arrangement of rhyme, and the location of tones. There are more than 800 rule sets for Ci, which is called Cipai [11]. The author of Ci needs to fill in the words according to the matrix associated to the Cipai. The uneven lines in Ci follow more continuous syntax than traditional Chinese Tang poetry [2].

1.3 Proposed Approach

Judy: To be extened We propose an AI system which generate Ci in an interactive approach. First, our system will prompt the user to provide a Cipai name. Because Ci belonging to different Cipai may contain different emotions or grammatical rules. Next, the system will receive few of keyword inputs that convey the detailed sentiments of the Song Ci. the first sentence of the iambic will be generated based on the keyword inputs. Further, the system generate following sentences based on previously-generated contexts using both RNN and SMT technique. Finally, we evaluate the quality of the generated Ci using an evaluation tool named BLEU.

1.4 Technical Challenges and Proposed Solutions

Judy: To be extened The first challenge to build a general model for all types of Song Ci. Different from Shi poetry whose structure is strict, Song Ci has more than 800 set of Cipai, and different Cipai follows different structural or rhythmic patterns. Therefore, it is difficult to generalize a model for all the Song Ci from limited training dataset. Our solution is to create a model based on Recurrent Neural Network. For every line generated in the SongCi, its probability is based on the probability of all previously lines.

Another challenge is to maintain consistent and poetic meanings throughout the generated SongCi. Compared with Shi poetry, Song Ci are much longer in length and therefore more complicated in context. It is difficult to keep long-distance memory using conventional RNN. Our solution is to use a Long Short Term Memory (LSTM) model that can track the long-distance information.

2. RELATED WORK

Judy: To be extened Approaches to poetry automatic generation can be divided into the following categories.

Using rules and templates. This approach adopts templates to generate poems that comply with grammatical rules, such as the rhythms, lines, and word frequencies [9,12].

Using evolutionary algorithms. This approach is mainly based on natural selection. It generates all possible candidates, and use search and evaluation algorithms to select the optimal one [6,7].

Using Statistical Machine Translation (SMT) methods. [5]. This approach first receives keywords and extract most relevant constituents to theses keywords. Next, it generates poems by iteratively selecting among these constituents based on phonological, structural, and poetic requiremetns.

Using neural network. This approach adopts an RNN Encoder-Decoder structure [1,10]. It generates new iambics context using previously-generated contexts. based on the rationale that, in Chinese poems, two consecutive lines have high semantical relevance.

3. METHODS

To automatically generate SongCi, First, we preprocess the SongCi corpus and tokenize each character. Then we use a vector space model to convert each Chinese character in the corpus to be a vector presentation in the vector space so that characters with similar semantic meanings have small distance in the vector space. Using the vector space as training data, we build a Recurrent Neural Network (RNN) that can generate SongCi with coherent and poetic meanings. We add Long short-term memory (LSTM) units in our RNN model to capture long-term semantic dependencies in Song Ci.

3.1 RNN

RNNs are the family of the deep learning structures to process sequential data [8]. Parameter sharing across the different parts of the model is the key idea that makes RNNs to be able to deal with the sequential data. However, a simple RNNs cannot learn long time dependency as in the optimization this term tends to vanish or explode very fast [3]. To solve this challenge, gated RNNs is proposed and becomes one of the most effective practical models that used for sequential data.

3.2 LSTM

Long short-term memory (LSTM) model [4] is one branch of such gated RNNs that is extremely successful in the application like speech recognition, machine translation, and handwriting generation. The key idea of LSTM is to introduce a self loop so that gradient can flow for long duration. The self loop (internal recurrence) is located in "LSTM cells" with outer recurrence like ordinary recurrent network. The weight of self-loop is controlled by a forget gate $f_i^{(t)}$:

$$f_i^{(t)} = \sigma(b_i^f + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)})$$

Where $\boldsymbol{x}^{(t)}$ is the current input vector and $\boldsymbol{h}^{(t)}$ is the current hidden layer vector, containing the outputs of all the LSTM cells. $\boldsymbol{b}^f$, $\boldsymbol{U}^f$, and $\boldsymbol{W}^f$ are biases, input weights, and recurrent weights of the forget gate, respectively. The internal state of LSTM cell is updated with the following equation:

$$s_i^{(t)} = f_i^{(t)} s_i^{(t-1)} + g_i^{(t)} \sigma(b_i + \sum_j U_{i,j}^f x_j^{(t)} + \sum_j W_{i,j}^f h_j^{(t-1)})$$

And the external input gate unit $g_i^{(t)}$ is computed with the following equation:

$$g_i^{(t)} = \sigma(b_i^g + \sum_j U_{i,j}^g x_j^{(t)} + \sum_j W_{i,j}^g h_j^{(t-1)})$$

The output $h^{(t)}$ and the output gate $q_i^{(t)}$, are updated using sigmoid function also:

$$\begin{aligned} h_i^{(t)} &= \tanh(s_i^{(t)}) q_i^{(t)} \\ q_i^{(t)} &= \sigma(b_i^o + \sum_j U_{i,j}^o x_j^{(t)} + \sum_j W_{i,j}^o h_j^{(t-1)}) \end{aligned}$$

LSTM is proven to be able to learn long-term dependencies more effectively than normal RNNs. In our project, we will use LSTM as our main method. We also plan to compare LSTM performance with other network structures.

4. DATA DESCRIPTION

Judy: To be filled ...

4.0.1 Tang Poetry Corpus

4.0.2 Song Ci Corpus

5. PROJECT MILESTONES

5.1 Completed Milestones

5.1.1 Background Survey

For this initial step, we plan to search for related works to computational literary creation to gain the basic knowledge of Song Ci. We are interested in the following questions: what is the criterion of a good Song Ci? How to evaluate the correctness, fluency and style of poems generated? Better understanding of related work and Song Ci composition rules will provide us with great help for the following work, especially algorithm testing and comparison.

5.1.2 *Corpus Search and Analysis*

The dataset we use contains 18668 Ci, which contains a total number of 1183 poets and 1170 Pai. This dataset basically covers Ci generated during the entire Song Dynasty and the beginning of Yuan Dynasty. We analyze the number of poems wrote by each poet, which shown in Figure 1. Most of the poems are created by the first poets. The one that creates the most poems is Qiji Xin, one of the most famous poet of the Southern Song Dynasty. His poems covers a wide range of styles. Among all those poems, the bold style is most will known by now. In addition, we statistic the Pai of each poem. Ci was first used as a lyric, and Pai is the name of the tune. Each Pai has a specific melody and rhythm, so Ci has a fixed format requirements, such as the number of sentences, the number of words per sentence, pronunciation of those words, rhyme and so on. The statistical result is shown in Figure **??**. The most popular Pai is Silk-Washing Stream, followed by Prelude To Water Melody, Partridge Sky, Pusaman and River of Red. And there is good reason to believe that the songs that corresponding to these Pai are beautiful in melody, lively in rhythm and easy to sung, which caused them to be so popular in ancient China.

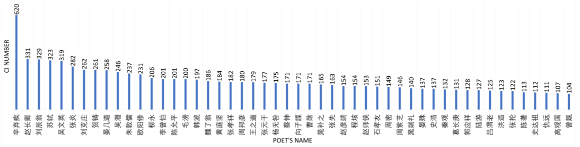


Figure 1: Poem Number Created by Top 50 Productive Poets

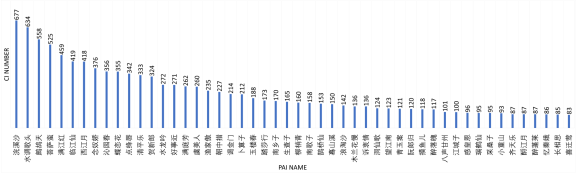
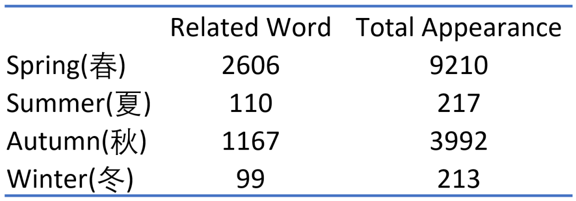


Figure 2: Poem Number of Top 50 Popular Pai

Word frequency analysis is to statistic and analyze the number of important words in the text, which is an important method of text mining. It is a traditional and useful content analysis method. The basic principle is to determine the overall style and theme of the entire article by the frequency of the words. By analyzing the word frequency in the poems, we have a general understanding of the style of poems and the process of writing those the poem, which can help us get more familiar with the grammar rules and themes of Ci. The most commonly used words can reveal common theme of Ci and the corresponding feelings. For example, we analyzed word frequency of season in our database. The result is shown in Figure 3. We found that spring related words reached 2606, these words appeared in our dataset for 9210 times. Followed by autumn, there are 1167 words associated with autumn and appeared 3992 times. The unique scene in spring and autumn can trigger people's emotions, which might be the reason that so many poems are related with these two seasons. From most frequently used words, shown in Figure **??**, we found that the moon, east wind, mortal world, wine, dream, rain, flowers , sunset, old friends are the most commonly used images. Commonly used places, including Jiangnan, West Lake, Changan, Fairy Isle, Yangzhou. Commonly used verbs including laugh , come back, go back, lovesickness, look back, meet by chance. Commonly used emotions are hate, worry, hard, sigh, desolate, haggard. These words represent a very broad theme and style of Ci, including the description of leaving and missing, pride and enthusiasm, seasonal terms, chanting things, chanting nostalgia and so on.



	Related Word	Total Appearance
Spring(春)	2606	9210
Summer(夏)	110	217
Autumn(秋)	1167	3992
Winter(冬)	99	213

Figure 3: Statistical Data of Season Related Word Frequency in Dataset

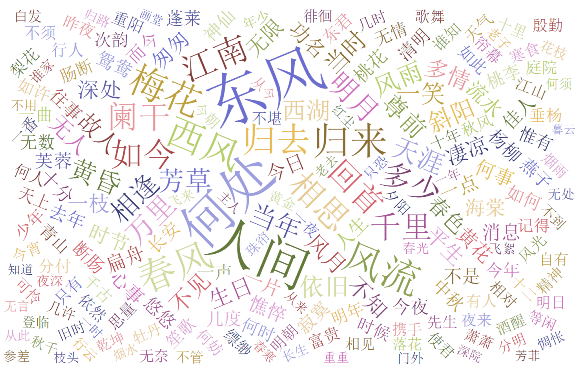


Figure 4: Wordcould of Frequently Used Words in Dataset

5.1.3 *Implementation of a Vector Space Model*

Vector space models (VSMs) represent words in a continuous vector space where semantically similar words are mapped to nearby points. We implemented this model to find the semantic relations between each Chinese character, so that given a few of keyword characters, such as 'spring' and 'beauty', we can generate poetries with coherent meanings using characters which are close to these keywords in the vector space. We give a visualized result in Figure 5. The figure is embeded with 100 Chinese characters in a 2-D space, which are randomly chosen from the most frequent 500 Chinese characters in the Song Ci corpus. The 2-D space corresponds to the first two dimensions in the vector space.

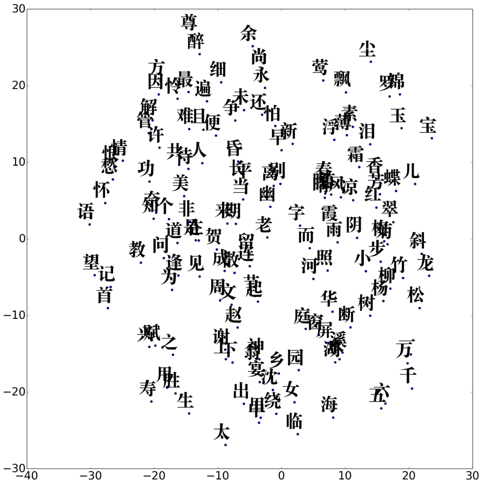


Figure 5: Vector Space Model

5.1.4 Implementation of a RNN + LSTM Model

We implement a preliminary version of our model. It is a special kind of RNN with Long Short Term Memory units (LSTM), which can capture long-term dependencies. The reasonale of applying LSTM to our model is that, different from Tang poetry, Song Ci has more content and is variant in styles. We may lose the coherent meaning of the generated Song Ci without backtracking previous context.

We present a prelimanry result in Figure 6.

5.2 Remaining Milestones

5.2.1 Implementation of Song Ci Generating Model

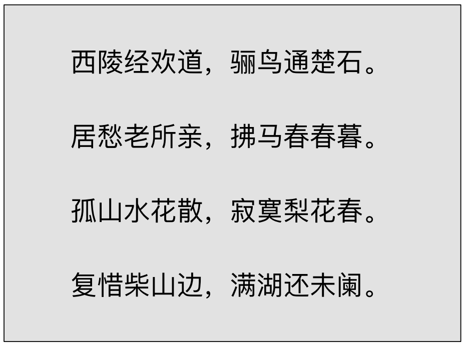
5.2.2 Model Testing and Comparison

5.2.3 Project Writing

6. REFERENCES

[1] D. Bahdanau, K. Cho, and Y. Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[2] Z.-q. Cai. *How to read Chinese poetry: a guided anthology*. Columbia University Press, 2008.



西陵经欢道，骊鸟通楚石。

居愁老所亲，拂马春春暮。

孤山水花散，寂寞梨花春。

复惜柴山边，满湖还未阑。

Figure 6: A poetry generated using RNN+LSTM model

[3] I. Goodfellow, Y. Bengio, and A. Courville. *Deep learning*. MIT Press, 2016.

[4] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[5] L. Jiang and M. Zhou. Generating chinese couplets using a statistical mt approach. In *Proceedings of the 22nd International Conference on Computational Linguistics-Volume 1*, pages 377–384. Association for Computational Linguistics, 2008.

[6] H. Manurung. An evolutionary algorithm approach to poetry generation. 2004.

[7] R. Manurung, G. Ritchie, and H. Thompson. Using genetic algorithms to create meaningful poetic text. *Journal of Experimental & Theoretical Artificial Intelligence*, 24(1):43–64, 2012.

[8] D. E. Rumelhart, P. Smolensky, J. L. McClelland, and G. Hinton. Sequential thought processes in pdp models. *Parallel distributed processing: explorations in the microstructures of cognition*, 2:3–57, 1986.

[9] N. Tosa, H. Obara, and M. Minoh. Hitch haiku: An interactive supporting system for composing haiku poem. In *International Conference on Entertainment Computing*, pages 209–216. Springer, 2008.

[10] Q. Wang, T. Luo, D. Wang, and C. Xing. Chinese song iambics generation with neural attention-based model. *arXiv preprint*

arXiv:1604.06274, 2016.

[11] Wikipedia. Ci (poetry). `https://en.wikipedia.org/wiki/Ci_(poetry)/`, last accessed 2017.

[12] X. Wu, N. Tosa, and R. Nakatsu. New hitch haiku: An interactive renku poem composition supporting tool applied for sightseeing navigation system. In *International Conference on Entertainment Computing*, pages 191–196. Springer, 2009.