iComposer: An Automatic Songwriting System for Chinese Popular Music

Hsin-Pei Lee
Institute of Information Science, Academia Sinica
hhpslily@iis.sinica.edu.tw

Jhih-Sheng Fan
Institute of Information Science, Academia Sinica
fann1993814@iis.sinica.edu.tw

Wei-Yun Ma
Institute of Information Science, Academia Sinica
ma@iis.sinica.edu.tw

Abstract

In this paper, we introduce *iComposer*, an interactive web-based songwriting system designed to assist human creators by greatly simplifying music production. *iComposer* automatically creates melodies to accompany any given text. It also enables users to generate a set of lyrics given arbitrary melodies. *iComposer* is based on three sequence-to-sequence models, which are used to predict melody, rhythm, and lyrics, respectively. Songs generated by *iComposer* are compared with human-composed and randomly-generated ones in a subjective test, the experimental results of which demonstrate the capability of the proposed system to write pleasing melodies and meaningful lyrics at a level similar to that of humans.

1 Introduction

Music is a universal language. There is no human society that does not, at some point, have a musical heritage and tradition. Over the past few years, many computer science researchers have worked on music information retrieval, focusing on genre recognition, symbolic melodic similarity, melody extraction, mood classification, etc.

With respect to computational creativity, tasks such as automatic music composition or lyrics generation have been discussed for decades. However, a relatively new topic—the relationship between lyrics and melody—remains somewhat mysterious. It is difficult to explain how music and spoken words fit together and why the pairing of these two creates an emotionally compelling experience; we believe this merits a deeper investigation.

Inspired by work on text-to-image synthesis and image caption generation, we propose *iComposer*, a simple and effective bi-directional songwriting system that automatically generates melody from text and vice versa. We believe that *iComposer* has value in capturing relationships between lyrics and melody. Moreover, *iComposer* makes it possible for more people, both professional and amateur musicians, to enjoy and benefit from this creative activity.

An LSTM (long short-term memory) based model is especially suitable for this task as it is structured to use historical information to predict the next value in the sequence. Our architecture consists of three subnetworks, each of which is a sequence-to-sequence model. These three subnetworks generate the pitch, duration, and lyrics for each note, and jointly learn the structure of Chinese popular music.

Choi, Fazekas, and Sandler (2016) use text-based LSTM for automatic music composition, Potash, Romanov, and Rumshisky (2015) demonstrate the effectiveness of LSTM on rap lyrics generation, and Watanabe et al. (2018) propose an RNN-based lyrics language model conditioned on melodies of Japanese songs. Bao et al. (2018) propose a sequence-to-sequence neural network model that composes melody from lyrics. However, to the best of our knowledge, *iComposer* is the first songwriting system that utilizes a sequence-to-sequence model in generating both melody and Chinese lyrics that match each other perfectly.

In addition to self-evaluation, we designed an experiment to evaluate the quality of the generated songs. Thirty subjects were asked to subjectively rate the aesthetic value of 10 selections, a mixture of original songs and computer-generated songs. The experimental results indicate that *iComposer* composes compelling songs that are quite similar to those composed by humans, and are much better than those generated by the baseline method.

2 Methodology

In this section, we briefly introduce our data gathering and pre-processing techniques, and then de-

Proceedings of NAACL-HLT 2019: Demonstrations, pages 84–88
Minneapolis, Minnesota, June 2 - June 7, 2019.

scribe the proposed network architecture and provide a system overview.

2.1 Dataset

For the purpose of this study, music sheets with vocal lines and their accompanying lyrics are necessary. Moreover, files with a single instrument corresponding to the melody are favored for better performance in the LSTM model. However, as far as we know there is no such dataset available, and most state-of-the-art melody extraction tools are still unsatisfactory.

We collected 1000 Chinese popular music pieces in MIDI format and extracted feature information using `pretty_midi`, a Python module for creating, manipulating, and analyzing MIDI files (Raffel and Ellis, 2014). Musical notes are represented by MIDI note numbers from 0 to 127, where 60 is defined as middle C. Note durations are represented by numbers in the range from 0.1 to 3.0 seconds.

We then recruited 20 people with at least five years of experience playing instruments to isolate the main melodies from polyphonic music and align the lyrics to their corresponding notes, as shown in Figure 1. The dataset consists of approximately 300,000 character-note pairs, of which we partitioned 80% as the training set and used the rest for testing. Note that the correspondence between notes and characters is not always one-to-one. A single sung character can be composed of multiple notes (i.e. one-to-many alignment).

1	我	以	為	要	是	唱	得	用	心	良	苦
2	74	76	78	76	74	73	69	74	73	74	69
3	妳	總	會	對	我	多	點	在	乎		
4	74	73	74	69	74	74	73	74	76		
5	我	以	為	雖	然	愛	情				
6	74	76	78	76	74	73	69				

1	我	以	為	要	是	唱	得	用	心	良	苦
2	0.3	0.3	0.6	0.3	0.3	0.6	0.6	0.6	0.3	0.3	1.2
3	妳	總	會	對	我	多	點	在	乎		
4	0.3	0.3	0.6	0.3	0.3	0.6	0.6	0.6	0.3		
5	我	以	為	雖	然	愛	情				
6	0.3	0.3	0.6	0.3	0.3	0.6	0.6				

Figure 1: Results of character-note alignment . Every character is bound to its corresponding note number and note duration.

2.2 Network Architecture

LSTM networks, introduced by Hochreiter and Schmidhuber (1997), have been shown effective for a wide variety of sequence-based tasks, including machine translation and speech recognition. They consist of a chain of memory cells that store state through multiple time steps to mitigate the vanishing gradient problem of recurrent neural networks.

The neural network proposed in this paper is a sequence-to-sequence model with a single layer and 100 nodes in both the encoder and decoder LSTM. Since music and lyrics both can be represented as a sequence of events, the generating process can be thought of as estimating the conditional probability

$$p(y_1, ..., y_n | x_1, ..., x_m) = \prod_{t=1}^{n} p(y_t | v, y_1, ..., y_{t-1})$$

where $x_1, \ldots, x_m$ is the input sequence, $y_1, \ldots, y_n$ is the output sequence, and v is the fixed-dimensional representation of the input sequence.

All the code for this system was written in Python using Pytorch as a backend, and was trained using stochastic gradient descent (batch size = 4), Adam optimization, and a learning rate of 0.001. The initial weights were randomly initialized with a range between -0.1 and 0.1.

2.2.1 Melody Generation

The melody is generated using two sequence-to-sequence models. The first predicts the note number, and the second predicts the note duration. As both models take the same text as input, it makes no difference which is run first. Once the pitch and duration of the notes are generated, we synthesize the data as audio using `pretty_midi`.

2.2.2 Lyrics Generation

In contrast to the melody generation process mentioned above, generating lyrics requires only one sequence-to-sequence model. We first extract note numbers from the given MIDI file, and then feed a sequence of pitches into the LSTM encoder. Then lyrics are automatically generated by the LSTM decoder.

3 Evaluation

In this section, we illustrate the behavior of the proposed system with an analysis of some generated lyrics and melodies. After this, we provide and discuss details about the subjective test.

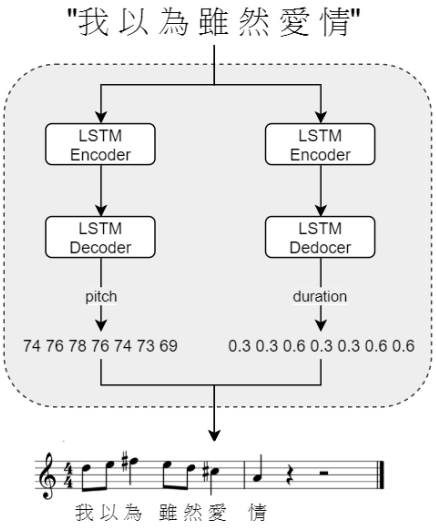


Figure 2: Melody generation model. Take the lyrics – "I thought that although love..." in Chinese as an example.

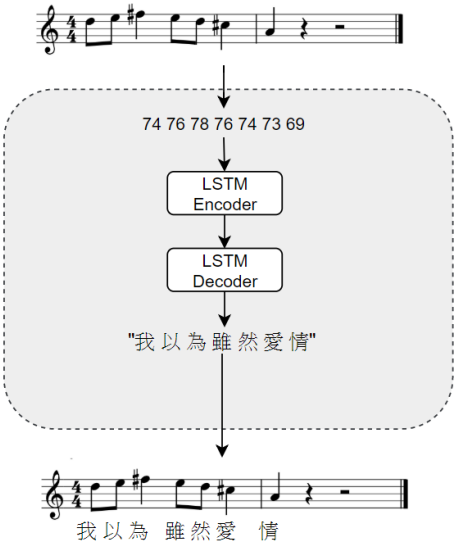


Figure 3: Lyrics generation model. Take the lyrics – "I thought that although love..." in Chinese as an example.

3.1 Analysis

The quality of the generated lyrics can be evaluated based on the variety of words. Here, we focus on this trait to show our model is learning and improving in a general sense.

According to our statistics on all the generated lyrics, only 1154 words are being used in epoch 10. In epoch 50, the variety of words increases to 1386. Finally, in epoch 100, there are 1646 different words in total. As the number of epoch increases, we could consistently see a noticeable improvement in lyrics.

Figure 4 presents some examples of generated lyrics from a certain melody. We can see that many words are repeated in the sequence in epoch 10. However, after epochs of training, the variety of words become greater. In epoch 100, there is no adjacent repeated words occur.

Epoch 10

```
Pride in your your
(Pride in your your)
我 們 都 不 想 聽 聽 說 話 來 說
(We don't want to hear what we say)
我 想 就 想 想
(I want to think about it)
```

Epoch 50

```
守 在 愛 裡 交
(Keep in love)
你 可 以 等 我 等 你 一 切
(You can wait for me to wait for you)
我 會 猜 向 你
(I will guess you)
```

Epoch 100

```
決 然 留 下 你
(Deciding to leave you)
水 向 東 流 時 間 怎 麼 留 慈 悲
(How to keep compassion when water flows eastward)
我 怕 會 聽 錯
(I am afraid I will get it wrong)
```

Figure 4: Lyrics generated from the same melody at different epochs. (Sentences in the parentheses are the English translation generated by Google Translation.)

3.2 Subjective Test

Since it is challenging to objectively evaluate the quality of songs, we evaluate the success of *iComposer* with experiments conducted using 30 human participants. All the subjects were college students aged 18 to 25, as the primary consumers of popular music are in this age group. In addition, we required our subjects to have as least five years of instrument playing or songwriting experience to ensure the reliability of our survey feedback.

The questionnaire contained 10 question groups, each with 3 melody clips or 3 sets of lyrics generated by either humans, *iComposer*, or the baseline model. These three types of songs are arranged in random order, and each is approximately 15 seconds in length. None of

the thirty chosen melodies and lyrics appeared in the training set, and were thus unseen to the model. In each case, an attempt was made to find less commonly known songs, so that the generated melodies and lyrics could be more fairly compared to the original ones.

For our baseline method, random notes were selected from MIDI note numbers 60–80, as pitch appears more frequently in this range according to the note distribution chart of our dataset shown in Figure 5 . The randomly-generated lyrics were chosen from 1000 most frequently used words in the dataset.

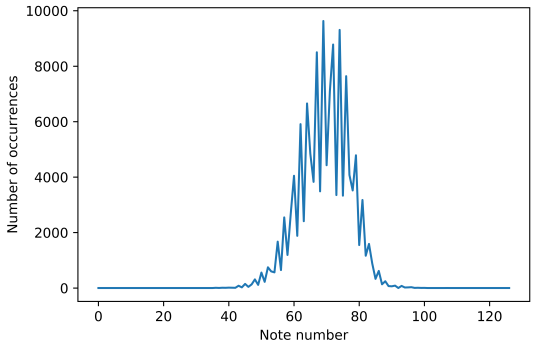


Figure 5: Frequency distribution of notes in the dataset

Survey participants were asked to listen to a mixture of melodies and lyrics generated by humans, *iComposer*, and our baseline model. It was not until the completion of the experiment that subjects were informed that some selections were computer-generated. During the experiment, subjects were asked to subjectively rate the melodies and lyrics from 1 to 10 in terms of the following standards (larger scores indicate better quality) :

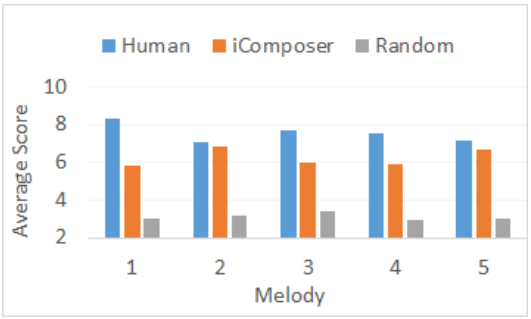
- **Melody**
  How smooth are the melodies?
- **Lyrics**
  How meaningful are the lyrics?
- **Overall**
  How well do the melodies fit with the lyrics?

Table 1: Subjective Test Results

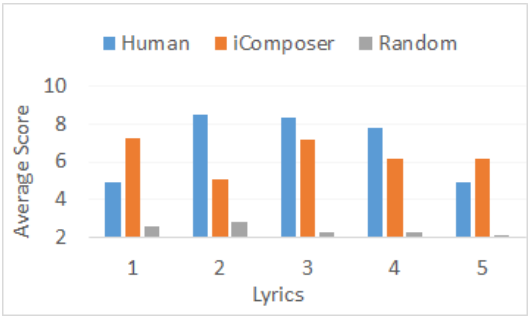
Model	Melody	Lyrics	Overall
baseline	3.12	2.43	3.24
iComposer	6.23	6.38	5.72
human	7.55	6.90	7.69

Table 1 shows the average responses to each question on the questionnaire mentioned above. According to the results, *iComposer* outperforms the baseline model in all three metrics, indicating its ability to generate songs in a more natural way. Moreover, melodies and lyrics generated by *iComposer* are rated slightly lower than human creations, which further suggests its effectiveness in songwriting. We can also observe from Table 1 that subjects give significantly lower scores to baseline model on **Lyrics** metrics. This may be due to the obvious grammatical errors the model made that make the rest two look a lot better. In contrast, an unexpected melody sometimes still demonstrate an acceptable level of melodic pleasantness and singability. Therefore, the gap between random-generated melody and the one created by *iComposer* is not that huge.

We also provide the average score of every question groups rated by 30 subjects. As shown in Figure 6 our model is capable of creating songs that are close to human creations in most cases. The model even produces better lyrics in lyrics set number 1 and 5.



(a) Average score of each melody



(b) Average score of each lyrics segment

Figure 6: Average aesthetic score of 15 melody clips and 15 sets of lyrics as rated by 30 human participants

4 Discussion

Our goal in this preliminary study is to build an artificial artist. The survey feedback illustrates that *iComposer* is promising.

Learning the structure of Chinese popular music is merely our first attempt: there are still a multitude of related issues that merit further exploration. For example, in Chinese the same syllable can be pronounced in four different tones: high, rising, rising then falling, and falling. Therefore, we could rate the fluidity of a song taking into account its conformity between flowing pitches and tones. Another novel task that interests us is the distinction between verse and chorus, namely, what particular features distinguish between verse and chorus. Taking these issues into account would dramatically improve our current work.

5 User Interface

iComposer is accessible via a web interface that allows users either to enter text data or to upload MIDI files. Once the text or melody has been given, users click the submit button to automatically compose songs.

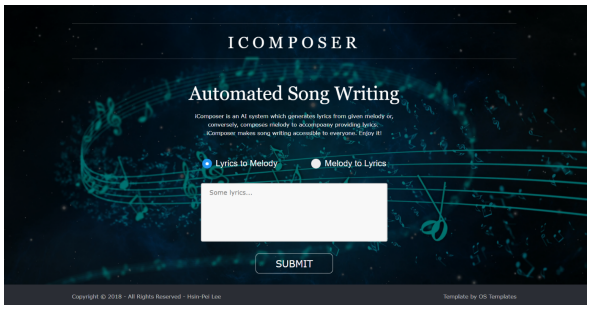


Figure 7: A web interface that allows users either to enter text data or to upload MIDI files.

After submission, users are directed to the music player interface. The lyrics shown on the page scroll down automatically in sync to the music. The system would play the music and also highlight the lyrics being sung. If users want to save the song, *iComposer* provides the songs in MIDI format for downloading.

Figure 8: Music player interface. It would play the music and highlight the lyrics being sung.

Our *iComposer* interface is accessible at: http://ckip.iis.sinica.edu.tw/service/iComposer

The demonstration video is posted on youtube at: https://www.youtube.com/watch?v=Gstzqls2f4A

The source code is available at: https://github.com/hhpslily/iComposer

References

Hangbo Bao, Shaohan Huang, Furu Wei, Lei Cui, Yu Wu, Chuanqi Tan, Songhao Piao, and Ming Zhou. 2018. Neural melody composition from lyrics. In *arXiv preprint arXiv:1809.04318*.

Keunwoo Choi, George Fazekas, and Mark Sandler. 2016. Text-based lstm networks for automatic music composition. In *Proceedings of the 1st Conference on Computer Simulation of Musical Creativity*.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Computation*, 9(8):1735–1780.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2015. Ghost-writer: Using an lstm for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924.

Colin Raffel and Daniel P. W. Ellis. 2014. Intuitive analysis, creation and manipulation of midi data with `pretty_midi`. In *Proceedings of the 15th International Society for Music Information Retrieval Conference Late Breaking and Demo Papers*, pages 84–93.

Kento Watanabe, Yuichiroh Matsubayashi, Satoru Fukayama, Masataka Goto, Kentaro Inui, and Tomoyasu Nakano. 2018. A melody-conditioned lyrics language model. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.