QiuNiu: A Chinese Lyrics Generation System with Passage-Level Input

Le Zhang, Rongsheng Zhang*, Xiaoxi Mao, Yongzhu Chang
Fuxi AI Lab, NetEase Inc., Hangzhou, China
{zhangle1, zhangrongsheng}@corp.netease.com

Abstract

Lyrics generation has been a very popular application of natural language generation. Previous works mainly focused on generating lyrics based on a couple of attributes or keywords, rendering very limited control over the content of the lyrics. In this paper, we demonstrate the *QiuNiu*, a Chinese lyrics generation system which is conditioned on passage-level text rather than a few attributes or keywords. By using the passage-level text as input, the content of generated lyrics is expected to reflect the nuances of users' needs. The *QiuNiu* system supports various forms of passage-level input, such as short stories, essays, poetry. The training of it is conducted under the framework of unsupervised machine translation, due to the lack of aligned passage-level text-to-lyrics corpus. We initialize the parameters of *QiuNiu* with a custom pretrained Chinese GPT-2 model and adopt a two-step process to fine-tune the model for better alignment between passage-level text and lyrics. Additionally, a postprocess module is used to filter and rerank the generated lyrics to select the ones of highest quality. The demo video of the system is available at https://youtu.be/OCQNzahqWgM.

1 Introduction

AI creation is an important application domain of Natural Language Generation (NLG), including story generation (Zhu et al., 2020; Alabdulkarim et al., 2021), poetry writing (Zhipeng et al., 2019; Liu et al., 2020; Yang et al., 2019), lyrics generation (Potash et al., 2015; Lee et al., 2019; Shen et al., 2019), etc,. Particularly, lyrics generation has always been a popular task of NLG since its intrusiveness and easy data availability. Previous works of lyrics generation (Castro and Attarian, 2018; Watanabe et al., 2018; Manjavacas et al., 2019; Fan et al., 2019; Li et al., 2020; Zhang et al., 2020) mainly focused on generating lyrics conditioned on specified keywords (e.g., *Flower*) or certain attributes such as the lyrics' text style (e.g., *Hip-hop*) and expected theme described by the lyrics (e.g., *Love*). However, these input only provide very limited control over the content of generated lyrics. Sometimes the generated lyrics may deviate far from the user's needs. To improve the usability of AI as a creation tool, we need to improve the controllability of the generated content.

Figure 1: This figure depicts a typical creation pattern: the author firstly conceives a rough draft (in the left box) and then polishes it to the final work (in the right box).

We argue that adopting free form text as the input is an approach to having precise control over the content of generated lyrics. As seen in Figure 1, an author usually conceives a passage (shown in the left text box) in his/her mind that expresses his/her inner feelings and thoughts, and then uses a wealth of writing skills and rhetorical techniques to create the final work (shown in the right text box).

In this paper, we demonstrate *QiuNiu* (the eldest son of the dragon in ancient Chinese mythology, who loves music), a Chinese lyrics generation system conditioned on free form passage-level text. The *QiuNiu* system can receive various forms of passage-level user input, which may be in different

* Corresponding Author

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
System Demonstrations, pages 76 - 82

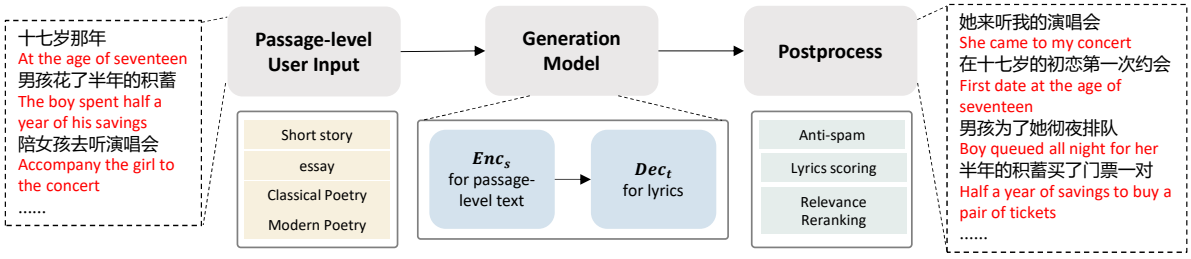

Figure 2: The architecture of *QiuNiu* system. The module of user input can receive various forms of passage-level text. And the generation model generates lyrics conditioned on the passage-level user input. Finally, a postprocess module is used to select the high-quality lyrics.

genres (e.g., short stories, essays, poetry), and eras (e.g., classical poetry, modern poetry). It is basically a text style transfer problem, which greatly suffers from the lack of aligned corpus. To construct the training data, we collected a passage-level corpus $\mathcal{D}_s$ from multiple sources and 300K different styles of lyrics corpus $\mathcal{D}_t$. Note that it is intractable to train a sequence-to-sequence (seq2seq) model from passage-level text to lyrics directly because the $\mathcal{D}_s$ and $\mathcal{D}_t$ are not aligned.

To address the issue, the *QiuNiu* system adopts the framework of unsupervised machine translation (UMT) (Lample et al., 2018; Yang et al., 2019). Specifically, The framework consists of an encoder Enc_s and a decoder Dec_s for the input side, an encoder Enc_t and a decoder Dec_t for lyrics side. The encoder Enc_s (or Enc_t) encodes the passage-level input text (or lyrics) into a hidden representation and the decoder Dec_s (or Dec_t) decodes it into lyrics (or passage-level text). The objective of the model training is to align the passage-level text and lyrics in the latent representation space.

To train the model, we first initialize the parameters with a custom Chinese GPT-2 (Radford et al., 2019) model, which is pretrained on around 30G Chinese books corpus collected online. Then we adopt a two-step process to finetune the model by jointly optimizing self-reconstruction loss, cross-reconstruction loss and alignment loss. After the training is finished, Enc_s encodes the passage-level input and Dec_t generates the candidate lyrics. Finally, a postprocess module is used to filter and rerank the generated lyrics to select the ones of highest quality. Human evaluation indicates the effectiveness of the framework.

The contributions of the *QiuNiu* system are summarized as follows:

1. The paper demonstrates the *QiuNiu* system, which can generate Chinese lyrics from various forms of passage-level text input for the first time.

2. To better align the passage-level text and lyrics, we propose a two-step process to finetune the UMT model of *QiuNiu*, which is initialized with the pretrained Chinese GPT-2 parameters. And a postprocess module is applied to select the high-quality lyrics by filtering and reranking the generated candidates.

3. The *QiuNiu* system and demo video are available at https://qiuniu.apps.danlu.netease.com/ and https://youtu.be/OCQNzahqWgM.

2 Architecture

The architecture of *QiuNiu* system is shown in Figure 2. It mainly consists of three modules: **Passage-level User Input**, **Generation Model** and **Postprocess**. Each module is described in detail below.

2.1 Passage-level User Input

The module receives passage-level inputs from the user, performs appropriate pre-processings and passes the results to the trained model to generate lyrics. A passage-level input here refers to a piece of text that can briefly depict the main idea that the lyrics is expected to convey. For the example in Figure 1, the author writes lyrics of lost love, which is based on the experiences of falling in love (e.g., "The boy spent half a year of his savings, accompany the girl to the concert.") and his own understanding of love (e.g., " Love comes and goes quickly, always leaves people in tears."). A passage-level text piece is much stronger than the keywords or attributes at depicting complex stories or nuanced feelings.

The *QiuNiu* system can support various forms of passage-level text inputs, such as short stories, essays, classical poetry, modern poetry. Though

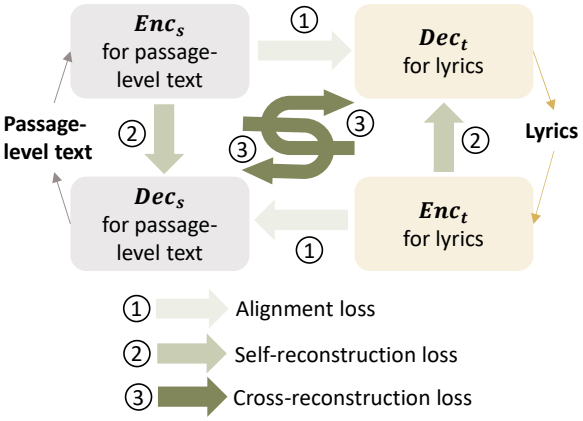

Figure 3: The framework of training the *QiuNiu* model. The framework is composed of two pairs of Encoder-Decoders, one pair for passage-level text and the other for lyrics. And the model is jointly optimized with the self-reconstruction loss, cross-reconstruction loss and alignment loss.

all these passage-level inputs have the same powerful semantic description capabilities, they may be different from each other in genres (e.g., story, poetry), and eras (e.g., classical text, modern text). In order to convert the input text into a form that can be processed by the generation model, preprocessings consists of conversion from traditional Chinese characters to simplified characters, spam filtering, error detection and correction, and conversion to token ids.

2.2 Generation Model

The generation model follows the Transformer-based sequence to sequence (seq2seq) framework (Vaswani et al., 2017), which consists of an Encoder for source text Enc_s and a Decoder for target text Dec_t. In the inference phrase, it takes in passage-level user inputs and generates several candidate lyrics. As shown in Figure 2, Enc_s encodes the passage-level text into latent representation and Dec_t decodes the latent representation into lyrics. We will describe the details of training below.

2.2.1 Corpus

Lyircs: We collected 300K different styles of Chinese lyrics from Internet, including Pop, Hip-hop, Chinese Neo-traditional, etc. For the lyrics corpus, we filtered the abnormal characters, removed lyrics less than 100 in length, and de-duplicated. We denote the processed lyrics corpus as $\mathcal{D}_s$.

Passage-level Text: To support various forms of passage-level text input, we collected the passage-level corpus covering different genres and eras from many sources. Specifically, the corpus contains short stories or essays collected from social medias, such as Weibo Tree Hole[1], Douban Essay[2], Micro Novel[3]. We filtered out the noisy text and processed them into uniform format. Besides, we also collected refined literature from both classical and modern eras which are naturally the passage-level text, mainly including Chinese classical poetry (e.g., Han Fu, Tang poetry, Song Ci, Yuan Qu, etc.), Chinese modern poetry with different styles (e.g., Philosophy, Love, Child, etc.). Finally, we obtain a passage-level corpus of 600K that denoted as $\mathcal{D}_t$.

Pseudo-aligned Dataset: Note that the passage-level text $\mathcal{D}_t$ and the lyrics $\mathcal{D}_s$ are not aligned. To help model for alignment, we further constructed a pseudo-aligned dataset $\mathcal{D}_a$, respectively for classical and modern text. For classical text, we first counted the n-gram ($n = 1, 2$) tokens in classical Chinese poetry of $\mathcal{D}_s$. Then for lyrics of each Chinese Neo-traditional song which is most similar to Chinese classical text, we selected these n-gram tokens appeared in lyrics and combined them based on the format of classical Chinese poetry (e.g., Five-character Quatrain, a four-line poetry with five characters each line). These pseudo poetry were finally paired with corresponding Chinese Neo-traditional lyrics. For modern text, We constructed pseudo-aligned pairs with back translation. Specifically, for lyrics of each song, we used the API[4] to first translate it into English text and then the English text was translated into Chinese plain text. Finally, we selected several segments of the translated plain text and reordered them, which is regarded as the aligned text with original lyrics.

2.2.2 Framework of Model

Due to the lack of aligned corpus from passage-level text to lyrics, we could not train the encoder and decoder of seq2seq model directly. Therefore, our training model adopts the framework of unsupervised machine translation (UMT) (Lample et al., 2018; Yang et al., 2019). As illustrated in Figure 3, the framework is composed of two pairs of Encoder-Decoders, one pair Enc_s-Dec_s for passage-level text and the other Enc_t-Dec_t for lyrics. Enc_s (or Enc_t) encodes passage-level text

[1]https://weibo.com/
[2]https://www.douban.com/
[3]https://www.567876.com/duanwen/weixiaoshuo/
[4]https://fanyi.youdao.com/

Method	Fluency	Coherence	Relevance	Overall Quality
Two-step Training	**3.05**	**2.86**	2.85	**2.98**
- step 1 (Reconstruction Loss only)	2.74	2.16	2.22	2.23
- step 2 (Alignment Loss only)	2.81	2.66	**3.06**	2.79

Table 1: Human evaluation results of Ablation.

(or lyrics) into latent representation, and Dec_s (or Dec_t) decodes the latent representation into passage-level text (or lyrics). The training object is to align the passage-level text and lyrics in the latent representation space.

Now we introduce three kinds of losses in the training process as shown in Figure 3.

1) **Alignment Loss:** The loss tries to capture the distribution of lyrics in Dec_t (or passage-level text in Dec_s) given the passage-level text in Enc_s (or lyrics in Enc_t). It optimizes the model parameters by calculating the negative log likelihood (NLL) on pseudo-aligned dataset $\mathcal{D}_a$:

$$\begin{aligned}\mathcal{L}_a = &-\sum_{\mathcal{D}_a} \log P\left(y_i | Dec_t\left(Enc_s\left(x_i\right)\right)\right) \\ &-\sum_{\mathcal{D}_a} \log P\left(x_i | Dec_s\left(Enc_t\left(y_i\right)\right)\right)\end{aligned} \tag{1}$$

where $(x_i, y_i) \in \mathcal{D}_a$ represent the pseudo-aligned passage-level text and lyrics respectively.

2) **Self-reconstruction Loss:** The loss is to calculate the reconstructed distribution for passage-level text or lyrics itself. Specifically, the passage-level text (or lyrics) is encoded into latent representation by Enc_s (or Enc_t) and then decoded by Dec_s (or Dec_t). The NLL loss is computed as

$$\begin{aligned}\mathcal{L}_{sr} = &-\sum_{x_{si} \in \mathcal{D}_s} \log P\left(x_{si} | Dec_s\left(Enc_s\left(x_{si}\right)\right)\right) \\ &-\sum_{x_{ti} \in \mathcal{D}_t} \log P\left(x_{ti} | Dec_t\left(Enc_t\left(x_{ti}\right)\right)\right)\end{aligned} \tag{2}$$

3) **Cross-reconstruction loss:** Given a passage-level text (or lyrics), we first generate lyrics (or passage-level text) by Enc_s-Dec_t (or Enc_t-Dec_s). Then the generated text is used to reconstruct the original input by Enc_t-Dec_s (or Enc_s-Dec_t). It is formulated as

$$\begin{aligned}\mathcal{L}_{cr} = &-\sum_{x_{si} \in \mathcal{D}_s} \log P\left(x_{si} | Dec_s\left(Enc_t\left(y^g_{ti}\right)\right)\right) \\ &-\sum_{x_{ti} \in \mathcal{D}_t} \log P\left(x_{ti} | Dec_t\left(Enc_s\left(y^g_{si}\right)\right)\right)\end{aligned} \tag{3}$$

where y^g_{ti} and y^g_{si} are the intermediate generated lyrics or passage-level text.

2.2.3 Training

Model Initialization: To make the model easier to learn and generate more fluent text, we first initialize the parameters of both the two encoder-decoder pairs with a pretrained GPT-2 model (Radford et al., 2019). Note that the encoders in our system use the unidirectional self-attention to be consistent with the structure of GPT-2. The pretrained GPT-2 with total 210 million parameters has 16 layers, 1,024 hidden dimensions and 16 self-attention heads. The GPT-2 is pretrained on about 30G Chinese novels collected online, whose vocabulary size is 11,400 and context size is 512.

Two-step Training: Next we use a two-step training method to finetune the model. In the first step, we train the Enc_s-Dec_t and Enc_t-Dec_s on constructed pseudo-aligned corpus $\mathcal{D}_a$ with alignment loss $\mathcal{L}_a$ for several epochs. Through this step, we improve the ability of the alignment between encoder and decoder, which is a warm-up for training on unaligned corpus. In the second step, the model is trained on all the corpus ($\mathcal{D}_a$, $\mathcal{D}_s$ and $\mathcal{D}_t$) with jointly optimizing the weighted alignment loss $\mathcal{L}_a$, self-reconstruction loss $\mathcal{L}_{sr}$ and cross-reconstruction loss $\mathcal{L}_{cr}$. In this step, The corpus $\mathcal{D}_s$ of passage-level text and $\mathcal{D}_t$ of lyrics is aligned in the latent representation space (Lample et al., 2018). In general, the training loss can be formulated as

$$\mathcal{L} = \alpha_1 \mathcal{L}_a + \alpha_2 \mathcal{L}_{sr} + \alpha_3 \mathcal{L}_{cr} \tag{4}$$

where $\alpha_1 = 1, \alpha_2 = 0, \alpha_3 = 0$ for the first step and $\alpha_1 = 1, \alpha_2 = 1, \alpha_3 = 1$ for the second step.

2.3 Postprocess

After the model training is finished, we use Enc_s and Dec_t to generate lyrics with the passage-level inputs in the inference phrase. Then we postprocess the candidates as followed.

Lyrics Scoring: To select the lyrics with high quality, we trained a classifier to judge whether the

Figure 4: The interface of the user input. Users can write multiple forms of passage-level text as input. Several examples of input are provided for each type.

candidate lyrics are good and use its confidence as the lyrics score $Score_l$. We used the lyrics of popular and classic songs as positive examples and the lyrics of less played songs as negative samples. The model is based on pretrained Chinese Bert (Devlin et al., 2019) implemented by Transformers[5]. Experimental results show that our model prefers to give high scores to graceful and ornate lyrics, such as metaphorical sentences, rather than the verbose and plain ones.

Relevance Reranking: The metric denoted as $Score_r$ is to measure the relevance between the passage-level inputs and the generated lyrics. The $Score_r$ is computed based on the n-gram ($n = 1, 2, 3, 4$) overlapping between the passage-level input $\mathcal{S}$ and the generated lyrics $\mathcal{T}$, which is denoted as $\mathcal{O}_n$. We formulate the $Score_r$ as follows:

$$Score_r = \frac{\sum_{n=1}^{N} O_n}{N\,|\mathcal{S}|}\,(N = 4) \tag{5}$$

where $|\mathcal{S}|$ is the length of passage-level input.

Finally, we rerank the lyrics filtered by an anti-spam process with the final score $Score_f$.

$$Score_f = Score_l + Score_r \tag{6}$$

3 Evaluation

3.1 Demonstration

In this section, we demonstrate how the *QiuNiu* system works. And mores details are described in the demo video.

The user input interface is shown in Figure 4. Users can choose one type of the passage-level text input, write passage-level text corresponding to the chosen type or try the provided examples as input. After that, click the button "*Generate!*".

Then we show some generated lyrics of different passage-level inputs, mainly including Chinese modern text and classical text.

1) Modern Text: The generated lyrics of two genres (short story, essay), as examples, are shown in the left and middle of Figure 5. For each genre, the *QiuNiu* system can perform well with content expansion and produce fluent and high-quality lyrics relevant to the inputs.

2) Classical Text: The *QiuNiu* system also supports Chinese classical poetry input. An example of Song Ci (a type of Chinese classical poetry) is shown in the right of Figure 5. Note that we can also receive other types of Chinese classical poetry, such as Tang poetry, Han Fu and so on. We will not show their generated results due to space limitation, but they are available at the url of *QiuNiu* https://qiuniu.apps.danlu.netease.com/.

3.2 Ablation

We conduct ablation study to evaluate our two-step training framework.

Metrics: We evaluate the generated lyrics from four perspectives: 1) *Fluency*: Is the lyric grammatically well formed, 2) *Coherence*: Is the lyric itself logical and consistent, 3) *Relevance*: Is the lyric relevant with the input, 4) *Overall quality*: Is the lyric a good lyric overall subjectively. Note *Overall quality* is not the average score of the others. All the metrics are scale from 1 to 5 while 5 is the best.

[5]https://github.com/huggingface/transformers

Figure 5: The examples of generated lyrics for different text inputs, including *short story*, *essay*, *classical poetry*.

Test set: We sampled 30 passage-level inputs for each of four genres (short story, essay, modern poetry and classical poetry), totally 120 samples.

Baselines: We compare our 1) *Two-step Training* method with 2) *Two-step Training - step 1* (use reconstruction loss only. Here we set $\alpha_1 = 0$ to remove the alignment loss) and 3) *Two-step Training - step 2* (use alignment loss only and can be considered as a seq2seq model with a small corpus).

We invited 3 evaluators to evaluate all the 120 generated lyrics generated independently. The results are shown in Table 1. All the scores are the means of 3*120 human evaluation results. *Two-step training* method gets around 0.2 promotion in perspectives of *Fluency*, *Coherence* and *Overall Quality*, which indicates the effectiveness. Reconstruction loss does make model acquire knowledge from more corpus and improve the fluency and coherence of the generated lyrics. The method *Two-step Training - step 2* achieve the best in *Relevance*. The supervised learning guarantees the correlation between the input and the generated lyrics while the unsupervised step slightly reduces the relevance. The method *Two-step Training - step 1* performs worst except in *Fluency*. This shows that the warm-up step is necessary for model to learn the connection between the input and lyrics.

4 Conclusion

In this paper, we demonstrate *QiuNiu*, a Chinese lyrics generation system conditioned on passage-level input. We support various forms of passage-level input, covering different genres and eras. The *QiuNiu* system adopts the framework of unsupervised machine translation due to the lack of aligned corpus from passage-level text to lyrics. Besides, the model of *QiuNiu* is initialized with the pre-trained Chinese GPT-2 parameters and finetuned in a two-step process to improve the alignment between the passage-level text and lyrics. Finally, a postprocess module is used to filter and rerank the generated lyrics to select the high-quality ones.

Acknowledgements

This work is supported by the Key Research and Development Program of Zhejiang Province (No. 2022C01011).

References

Amal Alabdulkarim, Siyan Li, and Xiangyu Peng. 2021. Automatic story generation: Challenges and attempts. *NAACL HLT 2021*, page 72.

Pablo Samuel Castro and Maria Attarian. 2018. Combining learned lyrical structures and vocabulary for improved lyric generation. *arXiv preprint arXiv:1811.04651*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT (1)*.

Haoshen Fan, Jie Wang, Bojin Zhuang, Shaojun Wang, and Jing Xiao. 2019. A hierarchical attention based seq2seq model for chinese lyrics generation. In *Pacific Rim International Conference on Artificial Intelligence*, pages 279–288. Springer.

Guillaume Lample, Myle Ott, Alexis Conneau, Ludovic Denoyer, and Marc'Aurelio Ranzato. 2018. Phrase-based & neural unsupervised machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5039–5049.

Hsin-Pei Lee, Jhih-Sheng Fang, and Wei-Yun Ma. 2019. iComposer: An automatic songwriting system for Chinese popular music. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics (Demonstrations)*, pages 84–88, Minneapolis, Minnesota. Association for Computational Linguistics.

Piji Li, Haisong Zhang, Xiaojiang Liu, and Shuming Shi. 2020. Rigid formats controlled text generation. *arXiv preprint arXiv:2004.08022*.

Yusen Liu, Dayiheng Liu, and Jiancheng Lv. 2020. Deep poetry: A chinese classical poetry generation system. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13626–13627.

Enrique Manjavacas, Mike Kestemont, and Folgert Karsdorp. 2019. Generation of hip-hop lyrics with hierarchical modeling and conditional templates. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 301–310.

Peter Potash, Alexey Romanov, and Anna Rumshisky. 2015. GhostWriter: Using an LSTM for automatic rap lyric generation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1919–1924, Lisbon, Portugal. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Liang-Hsin Shen, Pei-Lun Tai, Chao-Chung Wu, and Shou-De Lin. 2019. Controlling sequence-to-sequence models - a demonstration on neural-based acrostic generator. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 43–48, Hong Kong, China. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

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, Volume 1 (Long Papers)*, pages 163–172, New Orleans, Louisiana. Association for Computational Linguistics.

Zhichao Yang, Pengshan Cai, Yansong Feng, Fei Li, Weijiang Feng, Elena Suet-Ying Chiu, et al. 2019. Generating classical chinese poems from vernacular chinese. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6156–6165.

Rongsheng Zhang, Xiaoxi Mao, Le Li, Lin Jiang, Lin Chen, Zhiwei Hu, Yadong Xi, Changjie Fan, and Minlie Huang. 2020. Youling: an ai-assisted lyrics creation system. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 85–91.

Guo Zhipeng, Xiaoyuan Yi, Maosong Sun, Wenhao Li, Cheng Yang, Jiannan Liang, Huimin Chen, Yuhui Zhang, and Ruoyu Li. 2019. Jiuge: A human-machine collaborative chinese classical poetry generation system. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 25–30.

Yutao Zhu, Ruihua Song, Zhicheng Dou, NIE Jian-Yun, and Jin Zhou. 2020. Scriptwriter: Narrative-guided script generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8647–8657.