Postprint: Automatic Generation Method for Classical Chinese Poetry Based on Sequence-to-Sequence Neural Network Models

Authors: Huang Wenming, Wei Wancheng, Deng Zhenrong

Date: 2018-10-11T00:00:00+00:00

Abstract

Computer poetry generation is the first step toward achieving computer writing. Currently, computer poetry generation generally suffers from problems of unclear themes and inconsistency between the poem’ s content and the writing intent. To address these issues, emulating the process by which ancient poets composed poetry, we propose a method for generating classical Chinese poetry that is divided into two stages. The first stage obtains a poetry outline; this process employs the TextRank algorithm to extract keywords from user input text and proposes a sequence-to-sequence neural network model based on an attention mechanism for keyword expansion. The second stage generates each line of poetry based on the poetry outline; this process proposes a sequence-to-sequence neural network model incorporating a dual encoder and attention mechanism for classical poetry generation. Finally, the effectiveness of the proposed method is verified through evaluation of experimental results. Compared with baseline methods, the classical poetry generated by the proposed method exhibits more explicit thematic meaning and greater consistency between the content expressed in the poem and the writing intent.

Full Text

Preamble

Automatic Generation of Poetry Based on Sequence-to-Sequence Neural Network Model

Huang Wenming[1,2], *Wei Wancheng*[1], *Deng Zhenrong*[1,2]
[1]College of Computer & Information Security, Guilin University of Electronic Technology, Guilin 541004, China
[2]Guangxi Colleges & Universities Key Laboratory of Cloud Computing & Complex Systems, Guilin 541004, China

Abstract: Computer poetry generation represents the first step toward computational creative writing. Existing approaches often suffer from unclear thematic focus and inconsistency between poetic content and the author' s intended meaning. To address these limitations, this paper proposes a novel two-stage poetry generation method that emulates the traditional process of human poetry composition. The first stage constructs a poetic outline by extracting keywords from user input using the TextRank algorithm and extending them through an attention-based sequence-to-sequence neural network model. The second stage generates each line of poetry based on this outline using a sequence-to-sequence model with dual encoders and an attention mechanism. Experimental evaluations demonstrate the effectiveness of our approach, showing that poems generated by our method exhibit clearer thematic meaning and stronger alignment between content and writing intention compared to baseline methods.

Key words: keyword expansion; attention mechanism; sequence to sequence; neural network model; Chinese poetry generation

0 Introduction

Classical Chinese poetry represents the essence of Chinese culture, traditionally employed to extol heroic figures, beautiful landscapes, love, and friendship. Poetry is categorized into various forms such as Tang poetry, Song lyrics, and Yuan opera, each with distinct structural and prosodic constraints. Table 1 illustrates the most popular form—Tang quatrains. Quatrains impose strict limitations: each poem consists of four lines with either five or seven characters per line (five-character quatrains and seven-character quatrains, respectively). The tonal pattern of each character must be either "level" (平) or "oblique" (仄), with the final characters of the second and fourth lines sharing the same rhyme category [1]. These rigid structural and prosodic requirements create strong rhythmic qualities when recited.

Recent years have witnessed growing academic interest in automatic poetry generation. Researchers have explored diverse approaches, including rule and template-based methods [2–6], text generation algorithms [7–9], automatic summarization techniques [10], and statistical machine translation (SMT) [11,12]. More recently, deep learning methods have achieved remarkable success across various natural language generation tasks. For poetry generation specifically, studies [13–16] have framed the task as a sequence-to-sequence problem, generating the first line from user input and subsequently producing each subsequent line based on previously generated content. While these methods represent significant advances, they exhibit notable limitations: user input is often restricted to keywords or requires the first line to be provided manually, and the generated poems frequently display ambiguous themes and misalignment between content and writing intention.

The core issue stems from the fact that user intention only influences the first line generation, with minimal impact on the remaining three lines. This leads to thematic inconsistency and unclear poetic expression. To remedy these problems, this paper proposes a novel two-stage generation framework. Our contributions are threefold: First, we enable flexible user input by extracting keywords from unrestricted text—ranging from single words to entire paragraphs —using the TextRank algorithm. Second, we address thematic ambiguity by introducing, for the first time, an attention-based sequence-to-sequence model for keyword expansion, where the attention mechanism and bidirectional LSTM enhance interconnections among expanded keywords and capture poetic sentiment. Third, we improve content-intention alignment through a dual-encoder sequence-to-sequence model that generates each line strictly according to the outline, ensuring thematic coherence and consistency with the original writing intention.

2.1 Overview

This paper divides poetry generation into two distinct phases: outline construction based on user input, and complete poem generation from the outline. As illustrated in Figure 1 [Figure 1: see original paper], given user input, the first phase extracts keywords and expands them into N interconnected keywords (K_1, K_2, K_3, ···, K) that serve as the poetic outline. Each keyword K acts as a subheading for the i-th line. In the second phase, the model generates line L using K and all previously generated lines L_1: $_1$ as input, proceeding sequentially until the entire poem is completed. This approach ensures that each line is generated with explicit guidance from both the outline and the poetic context established by preceding lines.

2.2 Outline Construction

To generate a poem with N lines, we construct N semantically related keywords as an outline, with each keyword serving as the subheading for one line. The process begins by extracting keywords from user input A, which may vary in length. If A is lengthy, we extract the N most important keywords directly. If A is short and yields fewer than N keywords, we expand the set to reach N keywords.

2.2.1 Keyword Extraction

We employ the TextRank algorithm [22] to evaluate word importance within the input text. Derived from PageRank [23], TextRank is a graph-based ranking algorithm that constructs an undirected network where nodes represent words and edge weights reflect co-occurrence frequency. The algorithm iteratively computes node scores using the formula:

$$S(V_i) = (1-d) + d \times \sum_{V_j \in E(V_i)} \frac{\omega_{ji}}{\sum_{V_k \in E(V_j)} \omega_{jk}} S(V_j)$$

where ω_{ji} denotes the weight of the connection between nodes V_j and V_i, $E(V_i)$ represents the set of nodes connected to V_i, d is the damping factor (typically 0.8 [27]), and $S(V_i)$ is the TextRank score of node V_i initialized to 1.0. After convergence, we select the top M keywords (M $\leq$ N) ranked by their final scores.

2.2.2 Keyword Expansion

Since extracted keywords M are typically fewer than N, expansion is necessary. As the outline guides the poem' s theme and meaning, high-quality keyword expansion is crucial. We investigate three distinct approaches to achieve poetically meaningful expansion: (1) neural network language model-based expansion, (2) word2vec-based expansion, and (3) attention-based sequence-to-sequence expansion.

1) Neural Network Language Model-Based Expansion: We utilize a recurrent neural network language model (RNNLM) [19] with Gated Recurrent Units (GRU) [24] to capture long-term dependencies in sequential data. The model extends keywords using the formula $K_i = \text{RNNLM}(K_{1:i-1})$, where K_i is the i-th keyword and $K_{1:i-1}$ represents the preceding keyword sequence. Training data consists of keyword sequences extracted from classical poems, where TextRank identifies the most significant keyword from each line to form a sequence.

2) Word2vec-Based Expansion: The word2vec model represents words as real-valued vectors in a T-dimensional space, where vector similarity corresponds to semantic similarity. Trained on our poetry corpus with T = 100, this model identifies the word most similar to K_i in vector space as its expansion candidate.

3) Attention-Based Sequence-to-Sequence Expansion: We frame keyword expansion as a sequence-to-sequence problem, introducing for the first time an attention mechanism with bidirectional LSTM (BiLSTM) for this task. Our model, termed keseq2seq, treats the input sequence as extracted keywords plus all previously generated keywords, outputting a predicted keyword sequence. As shown in Figure 2 [Figure 2: see original paper], the encoder converts the input sequence $(X_1, X_2, ...)$ into hidden states $(h_1, h_2, ...)$, where X_i is the encoded vector of the i-th keyword. The decoder generates the output sequence $(y_1, y_2, ...)$, where each y_t is produced based on the previous output y_{t-1}, current state S_t, and context vector c_t. The context vector c_t aggregates encoder hidden states weighted by attention scores $a_{t,i}$, allowing the decoder to focus on relevant input words at each generation step. BiLSTM enhances this process by capturing both historical and future context, significantly strengthening inter-word relationships. While methods 1 and 2 are commonly used for keyword expansion, our experiments demonstrate that method 3 substantially outperforms them for poetry generation, as the attention-based model effectively learns connections between keywords across poetic lines, mimicking human associative thinking.

2.3 Poetry Generation Model

Poetry generation is also treated as a sequence-to-sequence task, but with two input sequences: the designated keyword and all previously generated lines. We enhance the standard attention-based sequence-to-sequence model [25] to support multiple input sequences, resulting in the dual-encoder poetry generation sequence-to-sequence model (pgseq2seq) shown in Figure 3 [Figure 3: see original paper].

For a keyword K with T_k characters and previously generated text X with T_x characters, the encoding stage employs BiLSTM to encode K into hidden state vectors $[h_1, ..., h_{T_k}]$ and X into $[h_1, ..., h_{T_x}]$. These are integrated into a unified vector r_c by concatenating the last forward state and first backward state of the BiLSTM:

$$r_c = \begin{bmatrix} \overrightarrow{h_{T_k}} \\ \overleftarrow{h_1} \end{bmatrix}$$

The combined hidden representation $H = [h_0, h_1, ..., h_{T_x}]$ is formed, where $h_0 = r_c$ represents the keyword and h_1 to h_{T_x} represent the generated text. When generating the first line, $T_x = 0$ and $H = [h_0]$, meaning the line is generated solely from the first outline keyword.

In the decoding stage, another LSTM generates the most probable output y_t at each time step based on the current state S_t, context vector c_t, and previous output y_{t-1}:

$$y_t = \arg\max_y P(y|y_{t-1}, S_t, c_t)$$

The state S_t is updated as:

$$S_t = f(S_{t-1}, y_{t-1}, c_t)$$

The context vector c_t is computed as a weighted sum of all input hidden states:

$$c_t = \sum_{j=0}^{T_x} a_{t,j} h_j$$

Attention weights $a_{t,j}$ are calculated by:

$$a_{t,j} = \frac{\exp(e_{t,j})}{\sum_{k=0}^{T_x} \exp(e_{t,k})}$$

where $e_{t,k}$ is computed as:

$$e_{t,k} = v_a^T \tanh(W_a S_{t-1} + U_a h_k)$$

Here v_a, W_a, and U_a are parameter matrices optimized during training. This dual-encoder architecture allows the model to simultaneously attend to both the outline keyword and the poetic context, ensuring generated lines align closely with the intended theme.

3.1 Data Processing

We collected 76,475 classical poems from the internet, randomly selecting 2,000 for validation and 2,000 for testing, with the remaining 72,475 used for training. All poems underwent word segmentation, after which TextRank scores were computed to identify the most significant word in each line as its keyword. This yielded 289,900 keywords from the training set, forming sequences as shown in Table 2 and Table 3 for training the keseq2seq model. For pgseq2seq training, each poem was formatted with its lines and corresponding keywords as illustrated in Table 4 .

3.2 Model Training

Both the keseq2seq model for keyword expansion and the pgseq2seq model for poetry generation were trained following Wang et al.'s [13] sequence-to-sequence training methodology. The training objective minimizes the cross-entropy loss between predicted and true data distributions using minibatch stochastic gradient descent, with the AdaDelta algorithm [25] adjusting learning rates. Model parameters were selected based on perplexity performance on the validation set. While pgseq2seq employs dual encoders, both models share identical encoder architectures.

3.3 Evaluation Method

Currently, no specialized automatic evaluation metric exists for poetry generation. Although BLEU has been adopted in prior work [13,26] and correlates moderately with human judgment, it cannot fully capture generation quality. Human evaluation remains the most effective assessment method. Following established practices [10–12], we evaluated poems across four dimensions: rhyme consistency, linguistic fluency, content coherence, and semantic meaning, each scored on a 5-point scale. We invited 20 scholars with master's degrees or higher to score 20 five-character quatrains and 20 seven-character quatrains generated by each system, with final scores averaged across all dimensions.

3.4 Experimental Results Analysis

We compared our approach against four baseline methods: SMT [11], RNNLM [27], RNNPG [12], and ANMT [23]. Table 5 presents the manual evaluation results, with Figures 4 [Figure 4: see original paper] and 5 [Figure 5: see original paper] visualizing the scores.

Our method outperformed all baselines for both five-character and seven-character quatrains. SMT surpassed RNNLM in rhyme consistency, confirming that translation-based approaches better capture inter-line rhyming patterns. ANMT performed better than SMT, RNNLM, and RNNPG but fell short of our approach. Both ANMT and our method utilize attention-based sequence-to-sequence architectures; however, our approach generates each line according to a pre-constructed outline. The comparison reveals modest improvements in rhyme and fluency but substantial gains in content coherence and semantic meaning. These improvements stem directly from our outline construction process, where the dual-encoder model receives both outline keywords and preceding lines as input, ensuring thematic consistency throughout the poem. Furthermore, the semantic connections among outline keywords significantly enhance the poem' s expressive meaning and thematic clarity, resulting in considerably higher average scores compared to ANMT and all other baselines.

3.5 Generation Examples

Table 6 presents two poems generated through human-computer interaction. For inputs "清明怀古" (Qingming Festival, reminiscing the past) and "看明月，思故乡" (gazing at the bright moon, missing home), the system first extracted keywords "清明；怀古" and "明月；故乡", then expanded them into outlines "清明；怀古；酒；萧然" and "明月；故乡；惆怅；感伤". The expanded keywords demonstrate clear semantic connections and emotional resonance with the original intent. The final poems vividly express nostalgic sorrow and homesick melancholy, respectively, confirming that our method produces thematically coherent and emotionally expressive poetry.

4 Conclusion

This paper proposes a two-stage approach to classical Chinese poetry generation. The first stage constructs a poetic outline of N interconnected keywords extracted from user input and expanded through an attention-based sequence-to-sequence model. The second stage generates N lines of poetry using a dual-encoder attention-based sequence-to-sequence model, where each line is guided by its corresponding outline keyword and previously generated content. Experimental results through manual evaluation demonstrate our method's superiority over existing baselines, achieving significant improvements in thematic clarity and content-intention alignment. This research contributes valuable insights to poetry generation and broader natural language generation tasks. Future work will incorporate topic models such as PLSA and LDA into the outline extraction stage and explore applications of our framework to other natural language generation domains.

References

[1] Li Wang. A summary of rhyming constraints of Chinese poems (Shi Ci Ge Lu Gai Yao) [M]. Beijing: Beijing Press, 2002.

[2] Naoko T, Hideto O, Michihiko M. Hitch haiku: an interactive supporting system for composing haiku poem [C]// Proc of Entertainment Computing. Berlin: Springer, 2008: 209-216.

[3] Wu Xiaofeng, Naoko T, Ryohei N. New hitch haiku: an interactive renku poem composition supporting tool applied for sightseeing navigation system [C]// Proc of Entertainment Computing. Berlin: Springer, 2009: 191-196.

[4] Yael N, David G, Goldberg Y, et al. Gaiku: generating haiku with word associations norms [C]// Proc of Workshop on Computational Approaches to Linguistic Creativity. Stroudsburg, PA: Association for Computational Linguistics, 2009: 32-39.

[5] Oliveira H G. Automatic generation of poetry: an overview [D]. [S. l.]: Universidade de Coimbra, 2009.

[6] Oliveira H G. PoeTryMe: a versatile platform for poetry generation [C]// Proc of ECAI Workshop on Computational Creativity, Concept Invention, and General Intelligence. 2012: 21.

[7] Ruli M, Graeme R, Henry T. Using genetic algorithms to create meaningful poetic text [J]. Journal of Experimental & Theoretical Artificial Intelligence, 2012, 24(1): 43–64.

[8] Manurung H M. An evolutionary algorithm approach to poetry generation [D]. [S. l.]: University of Edinburgh, 2004.

[9] Zhou Chengle, You Wei, Ding Xiaojun. Genetic algorithm and its implementation of automatic generation of chinese songci [J]. Journal of Software, 2010, 21(3): 427–437.

[10] Yan Rui, Jiang Han, Lapata M, et al. I, poet: automatic chinese poetry composition through a generative summarization framework under constrained optimization [C]// Proc of the 23rd International Joint Conference on Artificial Intelligence. 2013.

[11] He Jing, Zhou Ming, Jiang Long. Generating Chinese classical poems with statistical machine translation models [C]// Proc of the 26th AAAI Conference on Artificial Intelligence. 2012.

[12] Zhang Xingxing, Mirella L. Chinese poetry generation with recurrent neural networks [C]// Proc of Conference on Empirical Methods in Natural Language Processing. [S. l.]: Association for Computational Linguistics, 2014: 670-680.

[13] Wang Qixin, Luo Tianyi, Wang Dong, et al. Chinese song iambics generation with neural attention-based model [EB/OL]. (2016-06-21). https://arxiv.org/abs/1604.06274.

[14] Wang Zhe, He Wei, Wu Hua, et al. Chinese poetry generation with planning based neural network [C]// Proc of the 26th International Conference on Computational Linguistics. 2016: 1051-1060.

[15] Yi Xiaoyuan, Li Ruoyu, Sun Maosong. Generating Chinese classical poems with RNN encoder-decoder [EB/OL]. (2016-04-06). https://arxiv.org/abs/1604.01537.

[16] Greene E, Bodrumlu T, Knight K. Automatic analysis of rhythmic poetry with applications to generation and translation [C]// Proc of Conference on Empirical Methods in Natural Language Processing. Stroudsburg, PA: Association for Computational Linguistics, 2010: 524-533.

[17] Simon C, Jacob G, Tony V. Full-face poetry generation [C]// Proc of ICCC. 2012.

[18] Jiang Long, Zhou Ming. Generating chinese couplets using a statistical mt approach [C]// Proc of the 22nd International Conference on Computational Linguistics. Stroudsburg, PA: Association for Computational Linguistics, 2008: 377-384.

[19] Mikolov T, Karafiát M, Burget L, et al. Recurrent neural network based language model [C]// Proc of the 11th Annual Conference of the International Speech Communication Association. 2010: 3.

[20] Ghazvininejad M, Shi Xing, Choi Y, et al. Generating topical poetry [C]// Proc of Conference on Empirical Methods in Natural Language Processing. 2016.

[21] Ilya S, Oriol V, Le Q V. Sequence to sequence learning with neural networks [C]// Advances in Neural Information Processing Systems. 2014: 3104-3112.

[22] Rada M, Paul T. TexTrank: bringing order into text [C]// Proc of EMNLP. 2004.

[23] Brin S, Page L. The anatomy of a large-scale hypertextual Web search engine [J]. Computer Networks, 1998, 30: 107-117.

[24] Dzmitry B, Kyunghyun C, Yoshua B. Neural machine translation by jointly learning to align and translate [J]. arXiv preprint arXiv:1409.0473, 2014.

[25] Zeiler M D. Adadelta: an adaptive learning rate method [EB/OL]. (2012-12-22). https://arxiv.org/abs/1212.5701.

[26] 蒋锐滢, 崔磊, 何晶, 等. 基于主题模型和统计机器翻译方法的中文格律诗自动生成 [J]. 计算机学报, 2015, 38(12): 2426-2436. (Jiang Ruiying, Cui lei, He Jing, et al. Automatic generation of Chinese metrical poems based on thematic models and statistical machine translation methods [J]. Journal of Computer Science, 2015, 38(12): 2426-2436.)

[27] Graves A. Generating sequences with recurrent neural networks [EB/OL]. (2014-06-05). https://arxiv.org/abs/1308.0850.

Note: Figure translations are in progress. See original paper for figures.