He M, Chen Y, Zhao HK *et al.* Composing like an ancient Chinese poet: Learn to generate rhythmic Chinese poetry. JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 38(6): 1272-1287 Nov. 2023. DOI: 10.1007/s11390-023-1295-1

Composing Like an Ancient Chinese Poet: Learn to Generate Rhythmic Chinese Poetry

Ming He[1, 2, †] (何　明), Yan Chen[3, †] (陈　艳), Hong-Ke Zhao[4] (赵洪科)
Qi Liu[5] (刘　淇), *Senior Member, CCF, Member, IEEE*
Le Wu[6] (吴　乐), *Senior Member, CCF, Member, IEEE*, Yu Cui[7] (崔　羽), Gui-Hua Zeng[1, *] (曾贵华)
and Gui-Quan Liu[5] (刘贵全)

[1] *Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China*
[2] *AI Laboratory, Lenovo Research, Beijing 100193, China*
[3] *Baidu Incorporated, Beijing 100193, China*
[4] *College of Management and Economics, Tianjin University, Tianjin 300072, China*
[5] *School of Computer Science and Technology, University of Science and Technology of China, Hefei 230027, China*
[6] *School of Computer Science and Information Engineering, Hefei University of Technology, Hefei 230009, China*
[7] *Risk Control Department, OPay, Beijing 100010, China*

E-mail: heming4@lenovo.com; chenyan22@baidu.com; hongke@tju.edu.cn; qiliuql@ustc.edu.cn; lewu@hfut.edu.cn
yu.cui01@opay-inc.com; ghzeng@sjtu.edu.cn; gqliu@ustc.edu.cn

Received January 16, 2021; accepted September 18, 2023.

Abstract Automatic generation of Chinese classical poetry is still a challenging problem in artificial intelligence. Recently, Encoder-Decoder models have provided a few viable methods for poetry generation. However, by reviewing the prior methods, two major issues still need to be settled: 1) most of them are one-stage generation methods without further polishing; 2) they rarely take into consideration the restrictions of poetry, such as tone and rhyme. Intuitively, some ancient Chinese poets tended first to write a coarse poem underlying aesthetics and then deliberated its semantics; while others first create a semantic poem and then refine its aesthetics. On this basis, in order to better imitate the human creation procedure of poems, we propose a two-stage method (i.e., restricted polishing generation method) of which each stage focuses on the different aspects of poems (i.e., semantics and aesthetics), which can produce a higher quality of generated poems. In this way, the two-stage method develops into two symmetrical generation methods, the aesthetics-to-semantics method and the semantics-to-aesthetics method. In particular, we design a sampling method and a gate to formulate the tone and rhyme restrictions, which can further improve the rhythm of the generated poems. Experimental results demonstrate the superiority of our proposed two-stage method in both automatic evaluation metrics and human evaluation metrics compared with baselines, especially in yielding consistent improvements in tone and rhyme.

Keywords aesthetics, poetry generation, polishing, semantics, two-stage method

1 Introduction

As a part of splendid cultural heritages in human history, poetry is a higher-level form of linguistic expression with a high concentration of emotions, views, and stories. In strict accordance with formats, poets creating a highly concise format text to express their feelings and views need to satisfy both aesthetic and semantic constraints[1]. For example, regulated verse, the formal type of Chinese classical poetry, employs

Regular Paper

This work was supported by the National Natural Science Foundation of China under Grant Nos. 61922073 and 72101176.

†Co-First Author (Ming He and Yan Chen contributed equally to this work. Ming He mainly contributed to the method (RPG) designing and paper writing, and Yan Chen mainly contributed to the method (RPG) optimization and experiments realization.)

*Corresponding Author

Ping (level) and Ze (downward) tones and rhyme scheme to guarantee the aesthetics of poetry. Generally, quatrain (Jueju) and Lüshi are the most popular genres of Chinese classical poetry, although there are many genres[2]. Fig.1 shows an example of a widely-known quatrain in China. Compared with other literary genres, we can find a quatrain with three restrictions.

1) The quatrain consists of four lines, and each line has seven characters.

2) Each character has a particular tone (e.g., Ping or Ze), and the whole poem obeys a tonal pattern.

3) The last characters of the second and the fourth lines in a quatrain must belong to the same rhyme category.

With such strict restrictions, the well-written quatrain is considered full of rhythmic beauty[3]. That is why the poetry generation, especially for rhythmic Chinese poetry, is a hard task for people, let alone computers, compared with relevant tasks in modern languages.

However, on account of its aesthetics and cultural value, the automatic generation of Chinese classical poetry has drawn increasing public attention in the research community. Various types of methods are proposed by researchers, such as rules or templates based methods[4, 5], genetic algorithms based methods[6, 7], and statistical machine translation (SMT) methods[8], which can automatically generate poems but the quality of the generated poems still needs to be improved. Recently, deep neural networks based methods have gained significant research interest owing to their successful representation ability of natural language, and plenty of poetry generation methods based on Encoder-Decoder models are proposed[9–11]. Although these Encoder-Decoder based methods can generate Chinese classical poems with higher quality compared with previous methods (e.g., rules based methods[4, 5] and SMT methods[8]), they mainly focus on the semantics of poetry while ignoring the aesthetics. In addition, they usually assume poetry can be generated by one pass which is counterintuitive. As we know, in the real process of poetry creation, both the aesthetics and semantics of poetry should be polished in an iterative way. By professionally learning the domain knowledge on ancient poetry composition, we can draw two intuitive observations.

1) An elegant poem is rarely accomplished once. Generally, it may be modified many times.

2) Different stages have different focuses. Some poets have a tendency to write a coarse poem conformed to strict formats at first and then deliberate its semantics based on the whole draft, i.e., aesthetics first and then semantics; while others are apt to first create an underlying semantic poem and then refine its aesthetics, i.e., semantics first and then aesthetics.

Motivated by the two aforementioned observations, in this work, we argue that the Chinese classical poetry generation is not only learning the semantics of language, but also following the poetry aesthetic rules to ensure rhythmic beauty. Along this line, we propose a two-stage method, called the Restricted Polishing Generation method (RPG) to automatically generate Chinese classical poetry. In contrast to previous poetry generation methods, the proposed method has two decoders, in which the first-stage decoder generates a coarse line conformed to aesthetics (or semantics) and the second-stage decoder polishes the coarse line into a fine line satisfying semantics (or aesthetics). Because the focus of each stage is different, the two-stage method RPG develops into two

早发白帝城
The River Journey from White King City
作者：李白
Author: Bai Li
朝辞白帝彩云间，(P* P Z* Z Z P P)
Having Left White King Town at Dawn, Where Red Clouds Play,
千里江陵一日还。(Z* Z P P Z Z P)
I'm Back to Jiang Ling Through Sailing Long Miles One Day.
两岸猿声啼不住，(Z* Z P P P Z Z)
With Monkey's Cried Along Both Banks Still Ringing in My Ears,
轻舟已过万重山。(P* P Z* Z Z P P)
My Skiff Has Passed by Mountains, One and All, in Cheers.

Fig.1. Example of a widely-known quatrain. The tone is shown at the end of each line within the bracket. To be specific, P and Z represent the level tone and the downward tone, respectively, and * indicates that the tone can be either in the tonal pattern. The rhyming characters are highlighted in green.

symmetrical methods: the aesthetics-to-semantics method (AtoS) and semantics-to-aesthetics method (StoA).

Specifically, in AtoS, we design a sampling method (i.e., a hard switch to choose aesthetic rules) on the first-stage vocabulary distribution to generate an aesthetically coarse line, and then feed it into the second-stage decoder for further semantic polishing. Similarly, in StoA, to polish the semantic draft and make it rhythmic, the second-stage decoder employs a gate deduced from aesthetics as a soft switch to control final predictions. Thus, the two symmetrical methods (i.e., AtoS and StoA) of RPG can ensure that the generated poems are more aesthetic and semantic compared with existing studies. Experimental results also demonstrate the superiority of RPG in both automatic evaluation metrics and human evaluation metrics, especially in tone and rhyme. We believe the proposed RPG can shed new light on imitating the real process of poets' creation for poetry. The contributions of this paper are two-fold as follows.

- We devise RPG to generate Chinese classical poetry, which depicts poems independently and iteratively polishing aesthetics and semantics. At the same time, this method provides us with different insights into the poetry generation.
- We design a sampling method and a gate (corresponding to the hard and soft switches, respectively) to formulate the tone and rhyme restrictions, and further improve the quality of the generated poems.

The rest of this paper is organized as follows. Section 2 comprehensively describes the literature related to the poetry generation. Then, we introduce the proposed poetry generation method in Section 3 detailedly, including the Encoder-Decoder model, RPG, and two realizations of RPG, i.e., AtoS and StoA. In Section 4, we present the dataset, evaluation metrics (including the automatic and human evaluations), and the performance of our proposed RPG on the real dataset. Finally, we summarize this work and offer promising directions for future research in Section 5.

2 Related Work

Recently, poetry generation has attracted more and more researchers' attention, and various genre methods have been proposed. To be specific, the early kind of Chinese poetry generation is based on rules or templates[4, 5, 12–14]. For example, Oliveira and Cardoso[4] designed a platform for the automatic generation of poetry, which enables users to customize functions of general configurations, initial seed words, and poem generation methods. Particularly, Yan *et al.*[14] utilized an automatic summarization method to generate poems. Besides, genetic algorithms have also been adopted to improve the quality of generated poems[6, 7]. Manurung[6] proposed an evolutionary algorithm based method of poetry generation and considered the poetry generation as a state space search problem. It is noted that the goal state in the method is a text that meets three properties (i.e., grammar, meaningfulness, and poeticness), which further improves the quality of generated poems. One kind of the important poetry generation methods is based on the SMT method. For example, Jiang and Zhou[8] considered poetry generation as an SMT problem, in which the previous lines of poems are treated as the source language and translated into the next line. Furthermore, He *et al.*[2] generated four-line Chinese quatrains by expanding the SMT method.

With the development of computation and storage of computers, deep learning algorithms come into sight and are highly popular among academics owing to their successful representation of texts[15–23] and images[24–28]. In recent years, many neural networks based poetry generation methods have been proposed[11, 29, 30]. For example, Zhang and Lapata[11] first employed the recurrent neural network (RNN)[31] to generate Chinese poetry, where a character-based RNN language model is used to generate the first line after being given some input keywords and other lines sequentially according to a variant RNN. The authors of [3, 10, 32] treated the poem generation as an Encoder-Decoder problem and generated each line of the poem sequentially. To be specific, Wang *et al.*[3] first planned sub-topics by adopting keywords of poetry and then generated each line via encoding the keywords and poems together. Moreover, several methods[33–36] employ specific mechanisms (e.g., attention mechanism, working memory mechanism, and salient-clue mechanism) for Chinese poetry generation, which contribute to strengthening the relation between the encoders and the decoders, and improving the coherence between topics and meanings. Particularly, Yang *et al.*[36] adopted the conditional variational auto-encoder to generate Chinese poems to improve the quality of poetry with topic coherence. However, these methods mainly focus on the semantics of poetry while ignoring the aesthetics. Besides, these methods only adopt a straightforward one-stage generation, which is not appropriate for imitating the creation process of mankind.

A novel direction considers a new research topic of classical Chinese poetry generation in view of multi-modal information fusion (e.g., texts and images)[37–39]. To better combine visual information with semantic topic information, e.g., to generate Chinese poetry from images directly, Xu *et al.*[37] proposed an Encoder-Decoder based method with topic memory units. Particularly, the memory units can support the unlimited number of keywords extracted from images and determine a latent topic for each character in the generated poem. Besides, Liu *et al.*[38] designed a multi-modal (e.g., text, image, and theme) three-stage method for Chinese poetry generation by a hierarchy-attention seq2seq model, which can better capture character, phrase, and sentence information between contexts and improve the symmetry of generated poems. It is worth noting that the rhythmics of poetry is also studied in [1, 40, 41]. Specifically, Hopkins and Kiela[1] utilized a neural language model trained on phonetic encoding to learn an implicit representation of both the formats and contents of English poetry. And Ghazvininejad *et al.*[40] adopted neural language models with finite-state machinery to meet the tone and rhyme constraints. Similarly, these methods also cannot capture the characteristics of mankind's creation process ideally, which means that it is hard to ensure the quality of generated poems partly.

The above methods generate poems without any polishing or deliberating mechanism. In fact, several methods utilize the polishing procedure based methods to improve the quality of image generation[42] and text generation[43–48]. For example, Xia *et al.*[47] designed a deliberation method to generate the final sequence by the second decoder with an additional input of the generated sequence by the first decoder. And Zhang *et al.*[48] implemented a two-decoder deliberation procedure for text summarization by utilizing the transformer decoder. Particularly, there are several studies[49–51] perfecting the generated procedure with deliberating mechanism to refine the poetry generation. To be specific, i-poet[49] proposes a polishing framework by encoding the writing intents repetitively to refine the generated poems. Moreover, to jointly improve the coherence and novelty of Chinese poetry generation, Shen *et al.*[50] designed a novel method that could polish the generated drafts with a bidirectional sentence-level context and a refining vector. Followed by the deliberating mechanism, a recent work[51] devises an iterative polishing framework for Chinese poetry generation with high quality, which could polish the draft towards a higher level in the field of literalness and linguistics. In addition, there are some scholars employing style/sentiment information to generate poems[52, 53]. Compared with these methods, our method RPG further perfects the polishing process of poetry generation. In RPG, the polishing process includes two stages, and each stage's goal is different. In consideration of each stage's goal, we develop two symmetrical methods according to RPG, which can better imitate the poetry creation process. Besides, we employ the tone and rhyme rules to direct method learning to improve the quality of generated poems.

3 Method

In this section, we propose RPG for generating rhythmic Chinese classical poetry based on the inner information of the semantics and aesthetics, which can better imitate the poetry creation process of humans and further improve the quality of generated poems. To demonstrate our proposed RPG more clearly, we first describe the Encoder-Decoder model, and the overall architecture of RPG is illustrated in Fig.2. Next, we introduce details of two concrete realizations of RPG: AtoS and StoA.

3.1 Overview

We design RPG to depict poetry via independently polishing the aesthetics and semantics in an iterative way. To be specific, we design the two-stage poetry generation method based on the Encoder-Decoder model as shown in Fig.2. Formally, the method is composed of an encoder $\mathcal{E}$, and two decoders D_1 and D_2. Depending on the aesthetics or semantics in each stage, RPG develops into AtoS and StoA.

Practically, considering the N lines of a poem we generate, we formulate the i-th line's characters of the poem as LC_i. In this work, we use first $i-1$ lines' characters of the poem (denoted as $LC_{1\sim(i-1)}$) as inputs to generate LC_i of the poem, where $LC_{1\sim(i-1)} = LC_1||LC_2||\ldots||LC_{i-1}$ is concatenated with previous lines' characters from LC_1 to LC_{i-1} and $||$ is the concatenation operator. Taking LC_i's generation as an example, in the encoding stage, we encode $LC_{1\sim(i-1)}$

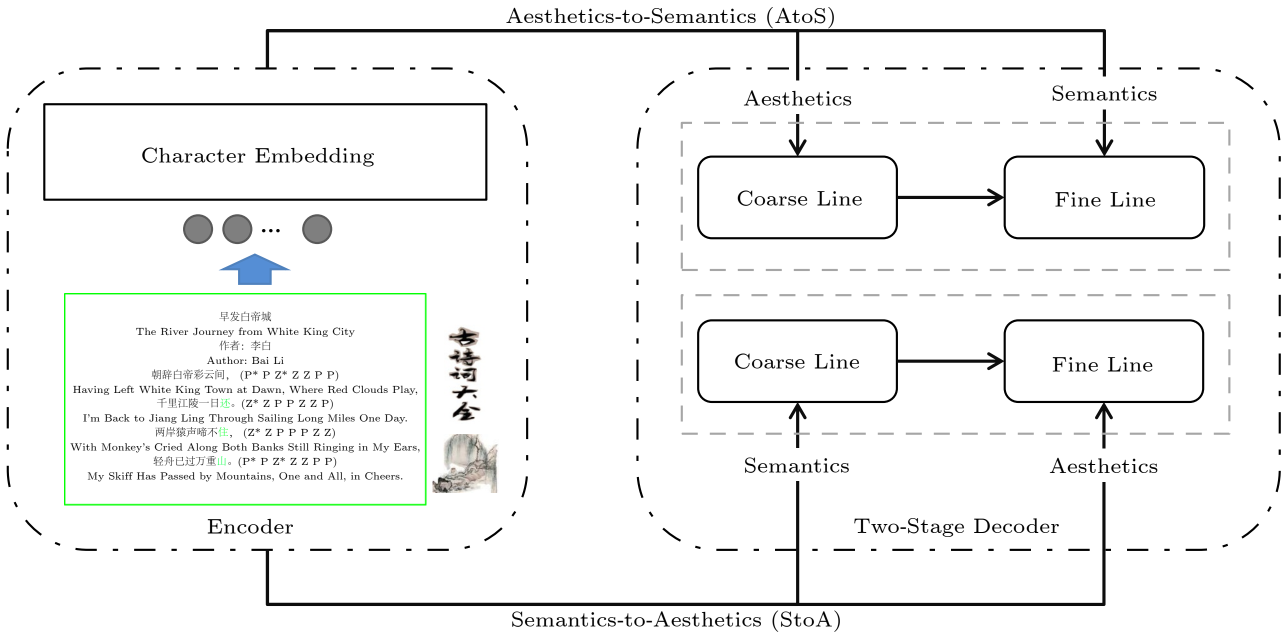

Fig.2. Proposed RPG's framework of poetry generation. AtoS first writes a coarse poem conformed to strict formats and then deliberates its semantics based on the whole drafts. StoA first creates an underlying semantic poem and then refines its aesthetics.

with vector $\boldsymbol{x}$ of $T_{\boldsymbol{x}}$ characters, where $\boldsymbol{x} = (\boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{x}_3, \ldots, \boldsymbol{x}_{T_{\boldsymbol{x}}})$. Then in the first-stage decoder D_1, we obtain a draft $\hat{\boldsymbol{y}}$ of $T_{\hat{\boldsymbol{y}}}$ characters based on encoding information, and the second-stage decoder D_2 generates the polishing line $\boldsymbol{y}$ of $T_{\boldsymbol{y}}$ characters:

$$\begin{aligned} \hat{\boldsymbol{y}} &= Sampling_{\hat{\boldsymbol{y}}}^{D_1}(Pr(\hat{\boldsymbol{y}}|\boldsymbol{x})), \\ \boldsymbol{y} &= Sampling_{\boldsymbol{y}}^{D_2}(Pr(\boldsymbol{y}|\hat{\boldsymbol{y}}, \boldsymbol{x})), \end{aligned} \quad (1)$$

where $Sampling$ is the representation of the sampling methods in each stage of the proposed method, $Pr(\ )$ is the decoding vocabulary distribution, $Pr(\hat{\boldsymbol{y}}|\boldsymbol{x})$ is the output of the first-stage decoder D_1, and $Pr(\boldsymbol{y}|\hat{\boldsymbol{y}}, \boldsymbol{x})$ is the output ot the second-stage decoder D_2. Finally, by concatenating all generated lines, we can obtain an elegant poem.

3.2 Encoder-Decoder Model

As demonstrated in Fig.3, $\mathcal{E}$ and D_1 can be treated as the Encoder-Decoder model. In the encoding stage (i.e., $\mathcal{E}$), given an input text $\boldsymbol{x}$ of $T_{\boldsymbol{x}}$ characters, the model encodes $\boldsymbol{x}$ into the hidden state $\boldsymbol{h}$, with a bi-directional Gated Recurrent Unit (GRU) model[54].

Specifically, $\boldsymbol{h}_i = BiGRU(\boldsymbol{x}_i, \boldsymbol{h}_{i-1})$, where $BiGRU$ is a sequence processing model with two GRUs, and $\boldsymbol{x}_i$ is the representation (e.g., word embedding vector) of the i-th word in $\boldsymbol{x}$. In the decoding stage (e.g., D_1), we use another GRU model, and it generates the hidden state $\hat{\boldsymbol{s}}$ and the output $\hat{\boldsymbol{y}}$ (e.g., sampling from the output to generate the coarse line), where $\hat{\boldsymbol{s}} = (\hat{\boldsymbol{s}}_1, \hat{\boldsymbol{s}}_2, \ldots, \hat{\boldsymbol{s}}_{T_{\hat{\boldsymbol{y}}}})$ and $\hat{\boldsymbol{y}} = (\hat{\boldsymbol{y}}_1, \hat{\boldsymbol{y}}_2, \ldots, \hat{\boldsymbol{y}}_{T_{\hat{\boldsymbol{y}}}})$. At the generation step $\hat{t}$, $\hat{\boldsymbol{y}}_{\hat{t}}$ is calculated as follows[9]:

$$\begin{aligned} \hat{\boldsymbol{y}}_{\hat{t}} &= \arg\max_{\boldsymbol{y}} Pr(\boldsymbol{y}|\hat{\boldsymbol{s}}_{\hat{t}}, \hat{\boldsymbol{A}}_{\hat{t}}, \boldsymbol{y}_{\hat{t}-1}), \\ Pr_{D_1} &= Pr(\hat{\boldsymbol{y}}_{\hat{t}}|\hat{\boldsymbol{s}}_{\hat{t}}, \hat{\boldsymbol{A}}_{\hat{t}}, \hat{\boldsymbol{y}}_{\hat{t}-1}) = g(\hat{\boldsymbol{s}}_{\hat{t}}, \hat{\boldsymbol{A}}_{\hat{t}}, \hat{\boldsymbol{y}}_{\hat{t}-1}), \end{aligned} \quad (2)$$

where $g(\cdot)$ is a nonlinear function (e.g., softmax).

After the previous character is generated, the hidden state $\hat{\boldsymbol{s}}_{\hat{t}}$ can be updated as the following:

$$\hat{\boldsymbol{s}}_{\hat{t}} = f(\hat{\boldsymbol{s}}_{\hat{t}-1}, \hat{\boldsymbol{A}}_{\hat{t}-1}, \hat{\boldsymbol{y}}_{\hat{t}-1}),$$

where $f(\cdot)$ is an activation function of GRU, and $\hat{\boldsymbol{A}}_{\hat{t}}$ is the attention vector that is recomputed at each step:

$$\hat{\boldsymbol{A}}_{\hat{t}} = \sum_{j=0}^{T_{\boldsymbol{x}-1}} \hat{\alpha}_{\hat{t}j} \boldsymbol{h}_j. \quad (3)$$

Besides, $\boldsymbol{h}_j$ is the j-th hidden state of the encoder $\mathcal{E}$, and the corresponding weight $\hat{\alpha}_{\hat{t}j}$ is computed by:

$$\begin{aligned} \hat{\alpha}_{\hat{t}j} &= \frac{\exp(e_{\hat{t}j})}{\sum_{k=0}^{T_{\boldsymbol{x}}-1} \exp(e_{\hat{t}k})}, \\ e_{\hat{t}j} &= \boldsymbol{V}_{\hat{\alpha}}^{\mathrm{T}} \tanh(\boldsymbol{W}_{\hat{\alpha}} \boldsymbol{s}_{\hat{t}-1} + \boldsymbol{U}_{\hat{\alpha}}^{\mathrm{T}} \boldsymbol{h}_j), \end{aligned}$$

where $e_{\hat{t}j}$ is the attention score on $\boldsymbol{h}_j$ at time step $\hat{t}$.

During the training, the loss function of the mod-

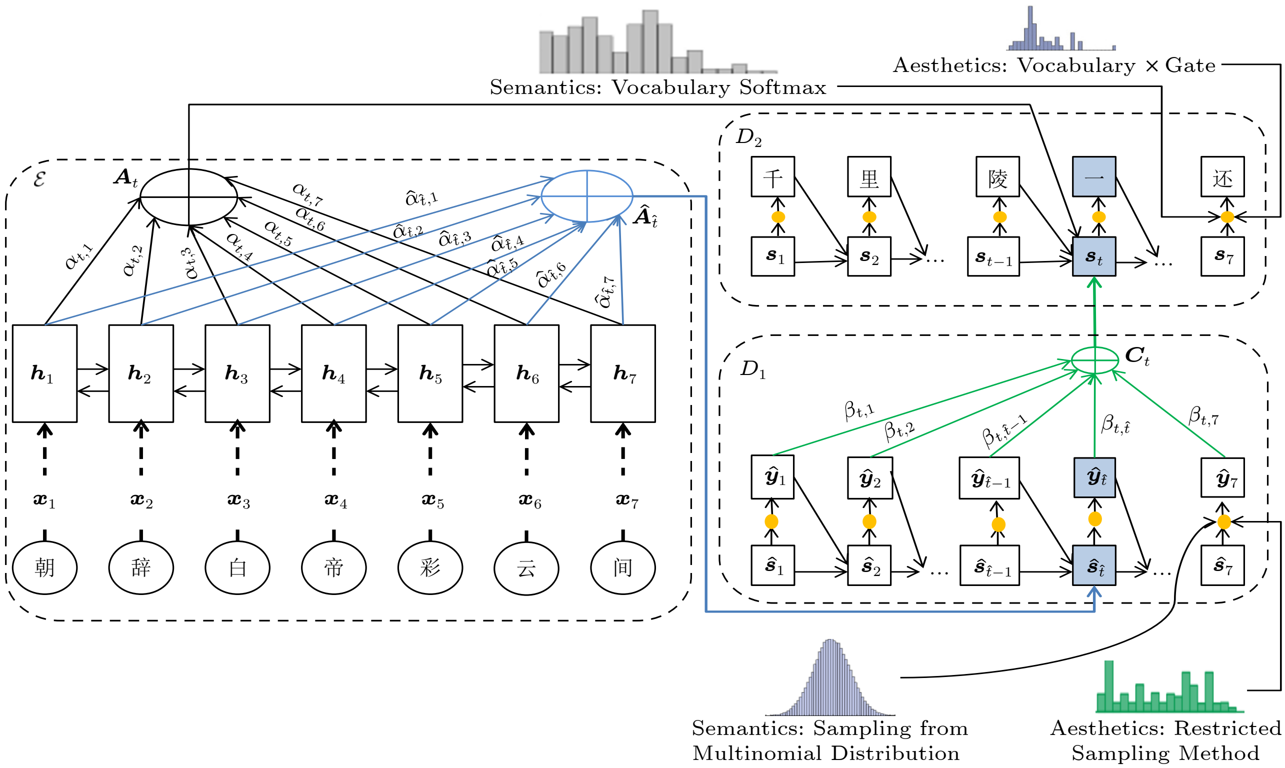

Fig.3. Illustration of RPG's architecture. The proposed method RPG includes two symmetrical methods: AtoS, with distributions of "Aesthetics: Restricted Sampling Method" and "Semantics: Vocabulary Softmax"; StoA, with distributions of "Semantics: Sampling from Multinomial Distribution" and "Aesthetics: Vocabulary Softmax × Gate".

el is the average negative log probability of the target word $\hat{\boldsymbol{y}}_{\hat{t}}$ at the generation step $\hat{t}$:

$$loss = -\frac{1}{N}\sum_{\hat{t}=1}^{N}\log(Pr(\hat{\boldsymbol{y}}_{\hat{t}}|\hat{\boldsymbol{s}}_{\hat{t}}, \hat{\boldsymbol{A}}_{\hat{t}}, \hat{\boldsymbol{y}}_{\hat{t}-1})),$$

where N is the sequence length. The loss is the cross entropy between the distribution of the true labels and the predictions given by the model. Therefore, by minimizing this loss, the weights are optimized to present precise distributions of true labels.

3.3 RPG

The proposed method RPG is composed of $\mathcal{E}$, D_1, and D_2. $\mathcal{E}$ and D_1 can be treated as the Encoder-Decoder model. To obtain the coarse line, we sample from the D_1 vocabulary distribution $Pr(\hat{\boldsymbol{y}}|\boldsymbol{x})$ in (1). In the second-stage decoder D_2, we feed the first-stage target sequence $\hat{\boldsymbol{y}}$ into D_2 for further polishing. As shown in Fig.3, D_2 outputs the fine line $\boldsymbol{y}$ of $T_{\boldsymbol{y}}$ characters based on encoder information, the coarse line $\hat{\boldsymbol{y}}$ and the hidden state $\hat{\boldsymbol{s}}$ generated by D_1. Specifically, differing from D_1, the output $\boldsymbol{y}_t$ of the second-stage decoder D_2 at the generation step t is defined as follows:

$$\boldsymbol{y}_t = Pr(\boldsymbol{y}_t|\boldsymbol{s}_t, \boldsymbol{A}_t, \boldsymbol{C}_t, \boldsymbol{y}_{t-1}),$$

where $\boldsymbol{s}_t = f(\boldsymbol{s}_{t-1}, \boldsymbol{A}_{t-1}, \boldsymbol{C}_{t-1}, \boldsymbol{y}_{t-1})$, and $\boldsymbol{A}_t$ is the attention vector between encoder $\mathcal{E}$ and the second-stage decoder D_2. Its definition is similar to $\hat{\boldsymbol{A}}_{\hat{t}}$ as shown in (3). D_2 has an attention vector $\boldsymbol{C}_t$ that can map the coarse line $\hat{\boldsymbol{y}}$ and the hidden state $\hat{\boldsymbol{s}}$ into a context vector, and $\boldsymbol{C}_t$ is defined as follows[47]:

$$\boldsymbol{C}_t = \sum_{j=0}^{T_{\hat{\boldsymbol{y}}}-1}\beta_{tj}[\hat{\boldsymbol{s}}_j; \hat{\boldsymbol{y}}_j],$$

$$\beta_{tj} = \frac{\exp(e_{tj})}{\sum_{k=0}^{T_{\hat{\boldsymbol{y}}}-1}\exp(e_{tk})},$$

where $e_{tj} = \boldsymbol{V}_{\beta}^{\mathrm{T}}\tanh(\boldsymbol{W}_{\beta}\boldsymbol{s}_{t-1} + \boldsymbol{U}_{\beta}^{\mathrm{T}}[\hat{\boldsymbol{s}}_j; \hat{\boldsymbol{y}}_j])$.

The generation probability of output $\boldsymbol{y}_t$ is computed as follows:

$$Pr(\boldsymbol{y}_t|\boldsymbol{s}_t, \boldsymbol{A}_t, \boldsymbol{C}_t, \boldsymbol{y}_{t-1}) = g(\boldsymbol{y}_t|\boldsymbol{s}_t, \boldsymbol{A}_t, \boldsymbol{C}_t, \boldsymbol{y}_{t-1}). \quad (4)$$

During the training, the training loss is trying to maximize the cross entropy between the predictions and the true labels[47]:

$$loss_1 = \frac{1}{N}\sum_{t=1}^{N}\log\sum_{\hat{\boldsymbol{y}}\in\mathcal{Y}}Pr(\boldsymbol{y}_t|\hat{\boldsymbol{y}}, \boldsymbol{x})Pr(\hat{\boldsymbol{y}}|\boldsymbol{x}), \quad (5)$$

where $Pr(\hat{\boldsymbol{y}}|\boldsymbol{x})$ is the output of the first-stage decoder D_1, and $Pr(\boldsymbol{y}_t|\hat{\boldsymbol{y}},\boldsymbol{x})$ is the output of the second-stage decoder D_2. Moreover, $\mathcal{Y}$ is the collection of all D_1's possible outputs, and is extremely hard to compute. Thus, Xia *et al.*[47] proposed a Monte Carlo-based method to optimize the lower bound of the loss in (5):

$$loss_1 = \frac{1}{N}\sum_{t=1}\sum_{\hat{\boldsymbol{y}}\in\mathcal{Y}} Pr(\hat{\boldsymbol{y}}|\boldsymbol{x}) \log Pr(\boldsymbol{y}_t|\hat{\boldsymbol{y}},\boldsymbol{x}). \tag{6}$$

We have demonstrated our proposed RPG in detail. Subsequently, we will present two concrete realizations (i.e., AtoS and StoA) of the proposed method RPG.

3.4 AtoS

AtoS first generates the coarse line under the restrictions of aesthetics and then refines the coarse line to a semantic one. As formulated in (1), to generate the draft $\hat{\boldsymbol{y}}$, we devise a restricted sampling method, which considers the aesthetic restrictions carefully. Specifically, we first design a tone-rhyme identification matrix to indicate the tone and rhyme of each word in the vocabulary. Then, in order to generate an aesthetic draft, we use the identification function to direct the final output. The identification matrix is adopted in (7) to sample $\hat{\boldsymbol{y}}_{\hat{t}}$:

$$Sampling_{\hat{\boldsymbol{y}}}^{D_1} = \text{multinomial}(\boldsymbol{I}\cdot Pr_{D_1}(\hat{\boldsymbol{y}}_{\hat{t}})), \tag{7}$$

where $Pr_{D_1}(\hat{\boldsymbol{y}}_{\hat{t}})$ is D_1 vocabulary distribution, and $\boldsymbol{I}$ is an information matrix with element 0 or 1 and is used to conduct the sampling operation following the rule, which is constructed by the tone-rhyme class (e.g., Ping or Ze) that each word belongs to, i.e., the element of $\boldsymbol{I}$ is equal to 1 if the character is allowed; otherwise, the element is equal to 0. Moreover, we adopt the multinomial distribution to sample the generated results (i.e., characters) based on the fusion of the identification function $\boldsymbol{I}$ and the output $Pr_{D_1}(\hat{\boldsymbol{y}}_{\hat{t}})$ of D_1, which can ensure the diversity of the generated output of D_1. Then, feeding the input sequence $\boldsymbol{x}$ and the output $\hat{\boldsymbol{y}}$ to the second decoder D_2, we sample from (4) to refine the coarse line $\hat{\boldsymbol{y}}$ to the fine line $\boldsymbol{y}$. In this paper, we adopt a beam search sampling method. During the training, we optimize parameters following the Monte Carlo algorithm in (6).

3.5 StoA

StoA is the other realization of the proposed two-stage method RPG, which first generates a coarse line underlying the semantics and then deliberates the coarse line to an aesthetic line. As depicted in (1), we first sample from the multinomial distribution of Pr_{D_1} formalized in (2). Then, we devise a gate in D_2 to control the final output that satisfies the aesthetics. Concretely speaking, we put the D_2 hidden state into a softmax layer to meet the tone and rhyme restrictions.

For each word i, the tone-rhyme gate is defined:

$$Pr_{1,i} = \text{softmax}(\boldsymbol{s}_i),$$

where $Pr_{1,i}$ is the rule (tone or rhyme) probability of word i, and $\boldsymbol{s}_i$ is the state of word i. More specifically, the rule is the tone or rhyme of word i.

Finally, we utilize the probability $Pr_{1,i}$ to restrict the final generated output $\boldsymbol{y}_i$, defined as $\boldsymbol{y}_i = Pr_{1,i}\cdot \boldsymbol{y}'_i$. It should be noted that $\boldsymbol{y}'_i$ is generated by D_2. Meanwhile, to train the method, the rule-based loss is defined as follows:

$$loss_2 = \frac{1}{N}\sum_{i=1}^{N}\log(Pr_{1,i}).$$

Hence, we maximize $loss_1 + loss_2$ to optimize the parameters of StoA.

4 Experiments

In this section, we first describe the real-world poem dataset. Then, we provide experimental settings, including baselines, implementation details, and evaluation metrics. Finally, we present the experimental results and conduct a case study to intuitively show the quality of poems polished by AtoS and StoA. It is worth noting that AtoS and StoA are two realizations of RPG. Therefore, the performance of AtoS or StoA represents the real performance of RPG.

4.1 Dataset

In our paper, we focus on the task of generating Chinese classical quatrains which have five or seven characters in each of the four lines. We collect 251 227 poems, crawled from mainstream poetry websites and processed by our experience (e.g., stored in a suitable format). Specifically, the dataset consists of 111 148 5-char poems and 140 079 7-char poems. Besides, each character in a poem has a specific tone (i.e., Ping or

Ze), and the last characters of the second and last lines in a quatrain must belong to the same rhyme category.

For the training, we randomly choose 3 000 poems for validation, 2 000 poems for test, and the rest for training. Then the RPG method generates poems line by line, which considers the previous lines as input to generate the current line, as exampled in Table 1.

Table 1. Training Lines Extracted from a Quatrain

Previous Line	Current Line
朝辞白帝彩云间	千里江陵一日还
朝辞白帝彩云间&千里江陵一日还	两岸猿声啼不住
朝辞白帝彩云间&千里江陵一日还&两岸猿声啼不住	轻舟已过万重山

4.2 Training Details

For the vocabulary, we choose 7 000 most frequently used characters as our vocabulary for the encoder, and the first-stage and second-stage decoders. Each character in the vocabulary is initialized by the word2vec model[55] and the embedding size is set to 512. UNK (short for Unknown) is used to replace the out-of-vocabulary characters. For the proposed RPG, GRU is adopted for the encoder and two decoders. Specifically, we employ BiGRU for the encoder, and two GRUs with different parameters for the two decoders, and all of the hidden sizes are set to 512. We first train the attention-based neural machine translation method (ANMT)[9] and then utilize its parameters to initialize the encoder and the first-stage decoder of AtoS and StoA. All the methods are optimized with the stochastic gradient descent (SGD) algorithm[56] and the learning rate is 0.5. The mini-batch size is fixed as 128 throughout all method training. The perplexity on the validation set is used for selecting parameters. In addition, AtoS and StoA are implemented on Tensorflow Framework and are trained on a single Tesla K80 GPU.

4.3 Comparison Methods

We adopt the following methods as baselines and use the same pre-processing procedure for all the methods as listed below.

- *SMT*. It generates a Chinese classical poetry by treating the previous lines as a source language and translating the source language (i.e., the previous lines) into the next line, which is a representative poetry generation method[2].
- *ANMT*. The attention-based neural machine translation method employs the Encoder-Decoder model with attention[9].
- *PPG* (*Planing-Based Poetry Generation*). This method first plans the sub-topics by adopting keywords of poetry and then generates each line by encoding the keywords and poems together based on the Encoder-Decoder model[3].
- *Delib* (*Deliberation-Based Method*). This method is firstly designed for machine translation, which also has two levels of decoders and adopts the deliberation mechanism in the Encoder-Decoder model[47]. Compared with this baseline Delib[47], it can intuitively demonstrate the effect of our proposed method RPG.
- *AtoS*. The aesthetics-to-semantics method has two stages of decoders. The first-stage decoder generates a coarse line conformed to the aesthetics, and the second-stage decoder polishes the coarse line to a fine line in consideration of the semantics.
- *StoA*. This is the other realization of our proposed method RPG. Conversely, the first-stage decoder generates a coarse line conformed to the semantics, and the second-stage decoder perfects the coarse line to a fine line met with the aesthetics.

4.4 Evaluation

We compare AtoS and StoA with the four aforementioned comparison methods (i.e., SMT[2], ANMT[9], PPG[3], and Delib[47]) in both human evaluation metrics and automatic evaluation metrics. Specifically, the human evaluation metrics are composed of four dimensions (i.e., poeticness, fluency, coherence, and meaning), and the automatic evaluation metrics are composed of bilingual evaluation understudy (BLEU)[57] and the tone-rhyme metrics.

4.4.1 Human Evaluation

First, we evaluate the generated poems by ancient poetry experts. This evaluation method is commonly used in poetry generation. Followed by related work[3, 11, 30], we evaluate the quality of generated poems through these metrics.

- *Poeticness*. The generated poem follows the tone and rhyme restrictions.
- *Fluency*. The poem reads smoothly and fluently.
- *Coherence*. The poem is coherent across lines.

• *Meaning*. The poem has a certain meaning and artistic conception.

We invite five experts to evaluate all the methods in these metrics with a rating from 1 (the worst score) to 5 (the best score). Particularly, to make a fair comparison, we shuffle all the methods' generated poems, i.e., the experts do not know each poem's source. It is worth noting that the experts are members of the China Traditional Culture Studies Association of Anhui Province, and they have extensive knowledge and expertise in Chinese classical poetry.

4.4.2 BLEU-Based Evaluation

Second, we adopt the metric BLEU as an automatic evaluation metric, which is popular in machine translation and can effectively evaluate the quality of machine-translated text from one natural language to another. Recently, it has been widely used in evaluating poetry generation[2, 11, 49]. In this paper, we adopt BLEU-1, BLEU-2, and BLEU-3 as our metrics to evaluate the performance of all the methods.

4.4.3 Tone-Rhyme Evaluation

As mentioned before, we generate Chinese classical poetry by considering the semantics and the aesthetics of poems simultaneously. It stimulates us to evaluate the tone-rhyme rules. Specifically, we design (8) and (9) to evaluate whether the method has the same tone-rhyme rules with the test set:

$$q_1 = \sum_{l=1}^{l=L} \frac{\sum_{i=1}^{T_l} q_{1,i}}{T_l}, \tag{8}$$

$$q_2 = \sum_{l=1}^{l=L} \frac{\sum_{i=1}^{T_l} q_{2,i}}{T_l}, \tag{9}$$

where L is the number of poems in the test set and T_l is the number of characters of each poem in the test set; $q_{1,i}$ and $q_{2,i}$ represent whether the tone and the rhyme of generations are the same as those of each character in real poems, respectively. If the i-th character's tone or rhyme of generations is the same with that of the corresponding character in real poems, $q_{1,i}$ or $q_{2,i}$ equals 1; otherwise 0.

4.4.4 Entropy and KL Divergence Evaluation

Neural Encoder-Decoder models tend to generate commonplace characters (e.g., the most-frequent characters in poems) regardless of the input. Intuitively, the energy of real poems is the highest among all the methods. The more the energy of the word distribution generated by the methods, the more diversified the methods. Also, the word distribution should be close to the true distribution. Hence, we adopt the metrics (i.e., entropy and KL divergence) to evaluate the generated quality of all the methods.

4.5 Results

Given the first line of a 5-char or 7-char real poem, all the methods generate the remaining lines. Besides being driven from a real poem, the first line also can be generated by some methods[3, 11]. The human and automatic evaluation results are exhibited as follows.

4.5.1 Human Evaluation Results

The human evaluation results are demonstrated in Table 2, and the bold value of the table is the best result of each column (so do all the other tables in this paper). We randomly choose 20 5-char poems and 20 7-char poems in the test set to evaluate.

Based on the results in Table 2, we can observe that our proposed AtoS and StoA perform the best. Specifically, the methods based on the Encoder-Decoder model perform better than the SMT method,

Table 2. Human Evaluation Results on 5-Char and 7-Char Quatrains

Method	Poeticness		Fluency		Coherence		Meaning		Average	
	5-Char	7-Char	5-Char	7-Char	5-Char	7-Char	5-Char	7-Char	5-Char	7-Char
SMT[2]	3.40	3.30	2.90	2.73	2.55	2.35	2.53	2.22	2.85	2.65
ANMT[9]	4.05	4.48	4.30	4.30	4.23	4.23	4.01	4.05	4.15	4.27
PPG[3]	4.28	4.55	4.30	4.18	4.20	4.23	3.78	3.98	4.14	4.24
Delib[47]	4.30	4.70	4.45	4.48	4.30	4.58	4.10	4.30	4.29	4.51
AtoS	**4.83**	4.70	**4.67**	**4.57**	**4.50**	4.57	**4.67**	4.33	**4.66**	4.54
StoA	4.63	**4.90**	4.57	4.50	4.33	**4.59**	4.30	**4.59**	4.45	**4.64**

which demonstrates the Encoder-Decoder model based methods can better capture the semantics of poems. PPG and ANMT perform almost the same and the deliberation method (i.e., Delib) is slightly better than them. It can indicate that the polishing process plays an important role in poetry generation. Particularly, AtoS and StoA are better than the deliberation method, which proves that our two-stage method RPG can better handle the poetry generation problem and generate higher-quality poems. And on 5-char poems, AtoS performs the best in the four metrics. Interestingly, StoA is slightly better than AtoS on 7-char poems generation. Besides, in poeticness, StoA and AtoS get almost full marks. This strongly indicates that RPG can better handle the tone and rhyme problems than baselines. Overall, RPG can generate a higher quality of Chinese poems compared with the baselines and better capture the semantics and aesthetics of poetry.

Besides, the poeticness result of StoA on 5-char is 4.63 and that on 7-char is 4.90, which means that the poeticness increases as the characters' length increases. This also demonstrates that the tone sampling performs better on longer sentences. Furthermore, the results of StoA and Delib (i.e., StoA on 5-char and 7-char is 4.63 and 4.90 respectively; Delib on 5-char and 7-char is 4.30 and 4.70 respectively) demonstrate that the sampling method plays a positive role in generating Chinese quatrains.

4.5.2 *BLEU-Based Results*

The BLEU-based evaluation results are demonstrated in Table 3. According to the results, we can observe that our methods AtoS and StoA significantly outperform the competitors in generating 5-char and 7-char quatrains. The results are relatively consistent with the human evaluation results as shown in Table 2. Specifically, ANMT and Delib have similar performance, and SMT is worse than the methods based on the Encoder-Decoder model. In the BLEU-1 evaluation results, AtoS achieves the best result against the others, PPG is the second-best, and StoA achieves the third when generating 5-char quatrains. As for generating 7-char quatrains, StoA performs the best. In the BLEU-2 evaluation results, AtoS outperforms all the other comparison methods on both 5-char and 7-char quatrains. In the BLEU-3 evaluation results, AtoS performs the best on 5-char quatrains while StoA achieves the best result on 7-char quatrains. In a word, we can observe that AtoS is good at generating 5-char quatrains, while StoA has its advantages in generating 7-char quatrains. Although PPG does a good job in BLEU-1 and BLEU-2, as the number of characters increases, PPG performs poorly.

On this basis, we can draw a conclusion that our proposed RPG has better control of the semantics.

Table 3. BLEU-Based Results on Quatrains

Method	BLEU-1		BLEU-2		BLEU-3	
	5-Char	7-Char	5-Char	7-Char	5-Char	7-Char
SMT	3.02	3.14	1.07	1.63	0.74	0.84
ANMT	6.50	6.32	2.50	2.08	1.21	0.82
PPG	7.33	7.55	2.79	2.53	0.83	0.91
Delib	6.51	6.23	2.46	2.04	1.24	0.85
AtoS	**7.64**	6.86	**3.31**	**2.56**	**1.65**	0.95
StoA	6.97	**7.58**	2.51	2.35	1.08	**1.04**

4.5.3 *Tone-Rhyme Results*

We report the tone-rhyme evaluation results as shown in Table 4.

Table 4. Tone-Rhyme Results on Quatrains

Method	q_1		q_2	
	5-Char	7-Char	5-Char	7-Char
SMT	0.53	0.54	0.07	0.14
ANMT	0.69	0.73	0.50	0.80
PPG	0.69	0.72	0.38	0.66
Delib	0.68	0.73	0.48	0.72
AtoS	**0.82**	**0.80**	**0.71**	**0.85**
StoA	0.69	0.75	0.52	0.82

The results also show that AtoS and StoA achieve the best performance and are relatively consistent with the above results. Specifically, AtoS is the best one, followed by StoA. SMT is worse than the Encoder-Decoder model based methods. The poems generated by our proposed RPG are more aesthetic, in yielding improvements in poeticness (tone and rhyme), which confirms the effectiveness of RPG. Note that, for AtoS and StoA, the results are different from human evaluations of poeticness. As a matter of fact, the designed metrics evaluating poeticness for models cannot fully reflect the true feelings on generated poems' poeticness of human experts, which is what makes the results different. In future work, we need to design more appropriate metrics to evaluate the performance in the tone and rhyme restrictions.

4.5.4 Entropy and KL Divergence Results

The entropy and KL divergence evaluation results are shown in Table 5. To understand all the methods' results more intuitive, we exhibit the entropy value of the truth distribution of real poems from the test dataset, denoted as "Truth" in Table 5. The results also reveal that AtoS and StoA outperform the baselines. Specifically, in entropy, AtoS performs the best on both 5-char and 7-char quatrains (e.g., AtoS on 5-char and 7-char is 6.17 and 6.09 respectively). In KL divergence, AtoS achieves the best performance on 5-char quatrains and StoA outperforms the others on 7-char quatrains. These observations show that our proposed RPG is more diverse and closer to the truth distribution.

Table 5. Entropy and KL Divergence Results

Method	Entropy		KL Divergence	
	5-Char	7-Char	5-Char	7-Char
SMT	5.64	5.75	0.38	0.42
ANMT	6.11	5.92	0.28	0.42
PPG	6.12	6.07	0.29	0.41
Delib	6.10	5.96	0.32	0.42
AtoS	**6.17**	**6.09**	**0.26**	0.36
StoA	6.14	6.02	0.27	**0.34**
Truth	6.75	6.80	0.00	0.00

In conclusion, the experimental results have shown that our proposed RPG delivers the best performance in both the automatic evaluation metrics and human evaluation metrics, especially in yielding consistent improvements in tone and rhyme, which strongly demonstrates the effectiveness of the two-stage polishing mechanism with different focuses and the prior knowledge adopted.

4.6 Examples

Table 6 and Table 7 show examples of the two-stage generation process for 5-char and 7-char poems by AtoS and StoA, respectively. Given the first line of a poem, AtoS and StoA generate the remaining lines. And combining all lines, we can get the whole poem. Obviously, the results show that poems generated in the D_2 stage are more semantic and aesthetic than those in the D_1 stage. For example, in Table 6, the poems generated in the D_1 stage are poetic but not fluent, as AtoS first deliberates the aesthetics and then refines the semantics. In Table 7, poems generated in the D_2 stage are more poetic than those in the D_1 stage. The reason is that D_2 can access the global information contained in the draft generated by D_1, and then D_2 can refine the draft under the restrictions of tone and rhyme. Therefore, D_2 can output a more poetic poem.

To better perceive the generated quality of AtoS and StoA, we develop a demo website[①]. On this website, users can input the first line (with a 5-char line

Table 6. Examples of AtoS

Stage	5-Char Poem	7-Char Poem
D_1	平生偏好酒, (*PPZZ) I like drinking,	一梦奢华去不还, (*ZPPZZP) Luxury dreams are gone,
	今日不胜**愁**. (*ZZPP) What a sad day.	空余残雪满空山. (*P*ZZPP) leaving the mountain with snow.
	一夜风吹起, (*ZPPZ) Wind blows all night,	如何不是长安道, (*P*ZPPZ) Why not take Chang'an road?
	满山云际**流**. (ZP*ZP) Cloud flows across mountains.	惟有青青白玉盘. (*ZPPZZP) Only the pale moon knows.
D_2	平生偏好酒, (*PPZZ) I like drinking,	一梦奢华去不还, (*ZPPZZP) Luxury dreams are gone,
	今日独登**楼**. (*ZZPP) I climbed on a building alone.	空余残雪满空**山**. (*P*ZZPP) leaving the mountain with snow.
	有酒能相醉, (*ZPPZ) Drinks can get me drunk,	可怜白发无人识, (*P*ZPPZ) No one recognizes the old man,
	无人可共**愁**. (PP*ZP) but no one can share my sorrow.	惟有孤云在目**间**. (*ZPPZZP) except the solitary cloud in his eyes.

Note: Red characters indicate the tone or rhyme is wrong.

① https://airecm.lenovo.com/common/poems/index.html, Oct. 2023. As this demo website is deployed on a non-free cloud server, this website is out of service in most of the time. If readers want to experience this demo, please send emails to heming01@foxmail.com, and we will turn on this demo temporarily.

Table 7. Examples of StoA

Stage	5-Char Poem	7-Char Poem
D_1	明月照秋叶, (*ZPPZ) Moon shines on the leaves, 孤舟横夕阳. (PPZZP) a boat travels in the sunset. 相思不相识, (*PPZZ) Someone I care is who I forget, 不复不相寻 . (*ZZPP) thus just let it go.	枫叶千枝复万枝, (*ZPPZZP) Maple leaves overlap on the branches, 秋风吹雨暗愁**思**. (*P*ZZPP) melancholy hides in the wind. 不知何处无人听, (*P*ZPPZ) I don't know how hide my feelings, 惟有梅花无限**时**. (*ZPPPZP) unless the spring lasts forever.
D_2	明月照秋叶, (*ZPPZ) Moon shines on the leaves, 孤舟泊暮**潮**. (PPZZP) a boat anchors for the night tide. 相望烟水外, (*PPZZ) I look through the water surface, 何处是渔**樵**. (*ZZPP) to find where is my destiny.	枫叶千枝复万枝, (*ZPPZZP) Maple leaves overlap on the branches, 秋风吹落鬓毛**丝**. (*P*ZZPP) autumn wind blew off my thining hair. 不知何处山中路, (*P*ZPPZ) Where was a path, 曾向西湖看月**时**. (*ZPPZZP) led me to the West Lake's moon.

Note: Red characters indicate the tone or rhyme is wrong.

or a 7-char line), and then the website can return the generated poems in real time according to the generated methods (i.e., AtoS and StoA). Finally, users can read the generated poems and perceive the generated quality of the poems intuitively. An example is given in Fig.4. Based on this real demo website, users can perceive the generated quality of poems by AtoS and StoA intuitively.

5 Conclusions

In this paper, motivated by the natural process of poet creation, we proposed RPG to generate Chinese classical poetry via polishing the aesthetics and semantics in an iterative way. Compared with existing methods, our proposed RPG can appropriately imitate the human procedure of poet creation and gener-

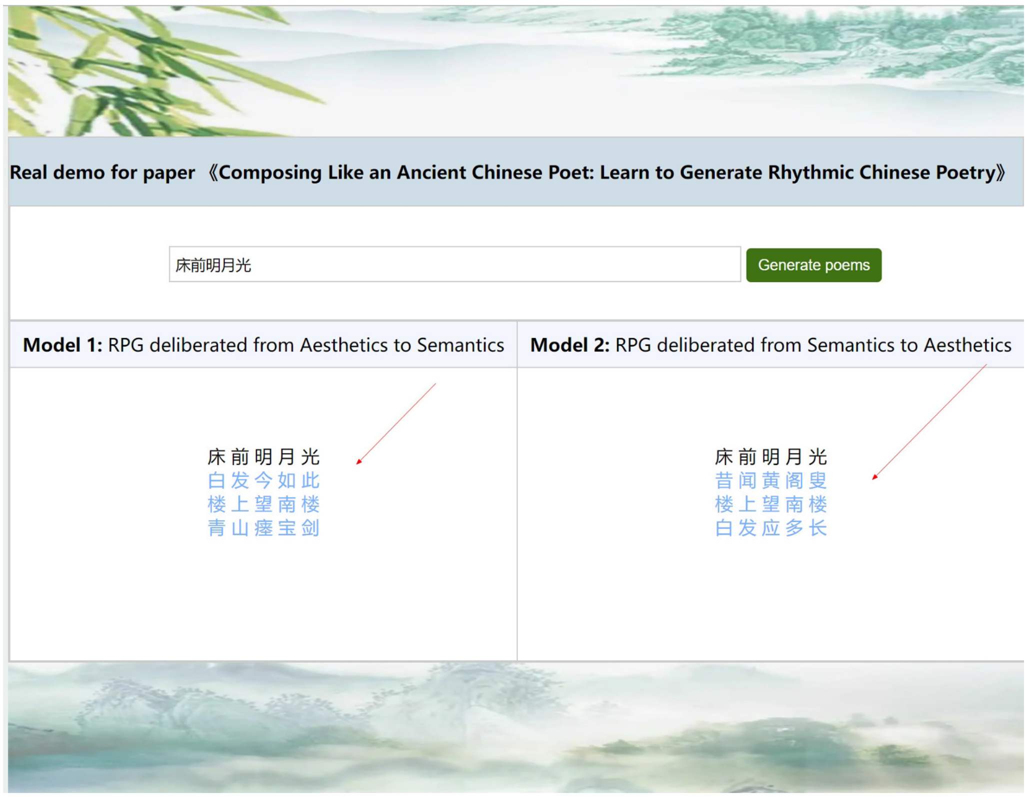

Fig.4. Example of generated poems.

ate higher-quality poems via considering the semantics and aesthetics of poems. Besides, the designed sampling method and the gate can further improve the quality of generated poems by formulating the tone and rhyme restrictions. The experimental results in the evaluation metrics (e.g., the human evaluation metrics, the BLEU-based metrics, and the tone-rhyme metrics) demonstrated that our proposed method RPG significantly outperforms the baselines. For example, AtoS achieved the best performance in the BLEU-2 metric (i.e., 3.31 on 5-char, 2.56 on 7-char). In future work, we will consider more restrictions (e.g., phonogram), design a self-adaption polishing process, and explore semi-structured or unstructured text generation by RPG.

Conflict of Interest The authors declare that they have no conflict of interest.

References

[1] Hopkins J, Kiela D. Automatically generating rhythmic verse with neural networks. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics*, July 2017, pp.168–178. DOI: 10.18653/v1/P17-1016.

[2] He J, Zhou M, Jiang L. Generating Chinese classical poems with statistical machine translation models. In *Proc. the 26th AAAI Conference on Artificial Intelligence*, July 2012, pp.1650–1656. DOI: 10.1609/aaai.v26i1.8344.

[3] Wang Z, He W, Wu H, Wu H Y, Li W, Wang H F, Chen E H. Chinese poetry generation with planning based neural network. arXiv: 1610.09889, 2016. https://arxiv.org/abs/1610.09889, Sept. 2023.

[4] Oliveira H G, Cardoso A. Poetry generation with PoeTryMe. In *Computational Creativity Research: Towards Creative Machines*, Besold T R, Schorlemmer M, Smaill A (eds.), Atlantis Press, 2015, pp.243–266. DOI: 10.2991/978-94-6239-085-0_12.

[5] Tosa N, Obara H, Minoh M. Hitch Haiku: An interactive supporting system for composing haiku poem. In *Proc. the 7th International Conference on Entertainment Computing*, Sept. 2008, pp.209–216. DOI: 10.1007/978-3-540-89222-9_26.

[6] Manurung H. An evolutionary algorithm approach to poetry generation [Ph.D. Thesis]. School of Informatics, The University of Edinburgh, Edinburgh, 2004.

[7] Manurung R, Ritchie G, Thompson H. Using genetic algorithms to create meaningful poetic text. *Journal of Experimental and Theoretical Artificial Intelligence*, 2012, 24(1): 43–64. DOI: 10.1080/0952813X.2010.539029.

[8] Jiang L, Zhou M. Generating Chinese couplets using a statistical MT approach. In *Proc. the 22nd International Conference on Computational Linguistics*, Aug. 2008, pp.377–384. DOI: 10.3115/1599081.1599129.

[9] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. arXiv: 1409.0473, 2016. https://arxiv.org/abs/1409.0473, Sept. 2023.

[10] Yi X Y, Li R Y, Sun M S. Generating Chinese classical poems with RNN encoder-decoder. In *Proc. the 2017 Conference on Chinese Computational Linguistics and Natural Language Processing Based on Naturally Annotated Big Data*, Oct. 2017, pp.211–223. DOI: 10.1007/978-3-319-69005-6_18.

[11] Zhang X X, Lapata M. Chinese poetry generation with recurrent neural networks. In *Proc. the 2014 Conference on Empirical Methods in Natural Language Processing*, Oct. 2014, pp.670–680. DOI: 10.3115/v1/D14-1074.

[12] Gervás P. An expert system for the composition of formal Spanish poetry. *Knowledge-Based Systems*, 2001, 14(3/4): 181–188. DOI: 10.1016/S0950-7051(01)00095-8.

[13] Wu X F, Tosa N, Nakatsu R. New hitch haiku: An interactive Renku poem composition supporting tool applied for sightseeing navigation system. In *Proc. the 8th International Conference on Entertainment Computing*, Sept. 2009, pp.191–196. DOI: 10.1007/978-3-642-04052-8_19.

[14] Yan R, Jiang H, Lapata M, Lin S D, Lv X Q, Li X M. i, Poet: Automatic Chinese poetry composition through a generative summarization framework under constrained optimization. In *Proc. the 23rd International Joint Conference on Artificial Intelligence*, Aug. 2013, pp.2197–2203. DOI: 10.5555/2540128.2540444.

[15] Jiang N, Chen J, Zhou R G, Wu C X, Chen H L, Zheng J Q, Wan T. PAN: Pipeline assisted neural networks model for data-to-text generation in social Internet of Things. *Information Sciences*, 2020, 530: 167–179. DOI: 10.1016/j.ins.2020.03.080.

[16] Ameur H, Jamoussi S, Hamadou A B. A new method for sentiment analysis using contextual auto-encoders. *Journal of Computer Science and Technology*, 2018, 33(6): 1307–1319. DOI: 10.1007/s11390-018-1889-1.

[17] Bengio Y, Ducharme R, Vincent P, Janvin C. A neural probabilistic language model. *The Journal of Machine Learning Research*, 2003, 3: 1137–1155. DOI: 10.1162/153244303322533223.

[18] Goldberg Y. Neural Network Methods for Natural Language Processing. Morgan and Claypool Publishers, 2017.

[19] Lai Y N, Zhang L F, Han D H, Zhou R, Wang G R. Fine-grained emotion classification of Chinese microblogs based on graph convolution networks. *World Wide Web*, 2020, 23(5): 2771–2787. DOI: 10.1007/s11280-020-00803-0.

[20] Li Y, Pan Q, Wang S H, Yang T, Cambria E. A generative model for category text generation. *Information Sciences*, 2018, 450: 301–315. DOI: 10.1016/j.ins.2018.03.050.

[21] Tao H Q, Tong S W, Zhao H K, Xu T, Jin B B, Liu Q. A radical-aware attention-based model for Chinese text classification. In *Proc. the 33rd AAAI Conference on Artificial Intelligence*, Jan. 2019, pp.5125–5132. DOI: 10.1609/aaai.v33i01.33015125.

[22] Yuan C Z, Bao Z F, Sanderson M, Tang Y. Incorporating word attention with convolutional neural networks for abstractive summarization. *World Wide Web*, 2020,

23(1): 267–287. DOI: 10.1007/s11280-019-00709-6.

[23] Zhu H Y, Liu Q, Yuan N J, Zhang K, Chen E H. Pop music generation: From melody to multi-style arrangement. *ACM Trans. Knowledge Discovery from Data*, 2020, 14(5): Article No. 54. DOI: 10.1145/3374915.

[24] Wu J J, Yang W, Li L D, Dong W S, Shi G M, Lin W S. Blind image quality prediction with hierarchical feature aggregation. *Information Sciences*, 2021, 552: 167–182. DOI: 10.1016/j.ins.2020.12.018.

[25] Jiang T B, Xia G S, Lu Q K, Shen W M. Retrieving aerial scene images with learned deep image-sketch features. *Journal of Computer Science and Technology*, 2017, 32(4): 726–737. DOI: 10.1007/s11390-017-1754-7.

[26] Liu H, Fu Z L, Han J G, Shao L, Hou S D, Chu Y Z. Single image super-resolution using multi-scale deep encoder-decoder with phase congruency edge map guidance. *Information Sciences*, 2019, 473: 44–58. DOI: 10.1016/j.ins.2018.09.018.

[27] Pawar K, Attar V. Deep learning approaches for video-based anomalous activity detection. *World Wide Web*, 2019, 22(2): 571–601. DOI: 10.1007/s11280-018-0582-1.

[28] Wu Y, Ma Y J, Liu J, Du J, Xing L. Self-attention convolutional neural network for improved MR image reconstruction. *Information Sciences*, 2019, 490: 317–328. DOI: 10.1016/j.ins.2019.03.080.

[29] Ghazvininejad M, Shi X, Priyadarshi J, Knight K. Hafez: An interactive poetry generation system. In *Proc. the ACL 2017, System Demonstrations*, July 2017, pp.43–48. DOI: 10.18653/v1/P17-4008.

[30] Wang Q X, Luo T Y, Wang D, Xing C. Chinese song iambics generation with neural attention-based model. In *Proc. the 25th International Joint Conference on Artificial Intelligence*, July 2016, pp.2943–2949. DOI: 10.5555/3060832.3061033.

[31] Mikolov T, Karafiát M, Burget L, Černocký J, Khudanpur S. Recurrent neural network based language model. In *Proc. the 11th Annual Conference of the International Speech Communication Association*, Sept. 2010, pp.1045–1048. DOI: 10.21437/Interspeech.2010-343.

[32] Wang Q X, Luo T Y, Wang D. Can machine generate traditional Chinese poetry? A feigenbaum test. In *Proc. the 8th International Conference on Brain Inspired Cognitive Systems*, Nov. 2016, pp.34–46. DOI: 10.1007/978-3-319-49685-6_4.

[33] Li J T, Song Y, Zhang H S, Chen D M, Shi S M, Zhao D Y, Yan R. Generating classical Chinese poems via conditional variational autoencoder and adversarial training. In *Proc. the 2018 Conference on Empirical Methods in Natural Language Processing*, Oct. 31–Nov. 4, 2018, pp.3890–3900. DOI: 10.18653/v1/D18-1423.

[34] Yi X Y, Li R Y, Sun M S. Chinese poetry generation with a salient-clue mechanism. In *Proc. the 22nd Conference on Computational Natural Language Learning*, Oct. 2018, pp.241–250. DOI: 10.18653/v1/K18-1024.

[35] Zhang J Y, Feng Y, Wang D, Wang Y, Abel A, Zhang S Y, Zhang A D. Flexible and creative Chinese poetry generation using neural memory. In *Proc. the 55th Annual Meeting of the Association for Computational Linguistics*, July 2017, pp.1364–1373. DOI: 10.18653/v1/P17-1125.

[36] Yang X P, Lin X W, Suo S D, Li M. Generating thematic Chinese poetry using conditional variational autoencoders with hybrid decoders. In *Proc. the 27th International Joint Conference on Artificial Intelligence*, July 2018, pp.4539–4545. DOI: 10.24963/ijcai.2018/631.

[37] Xu L L, Jiang L, Qin C, Wang Z, Du D F. How images inspire poems: Generating classical Chinese poetry from images with memory networks. In *Proc. the 32nd AAAI Conference on Artificial Intelligence*, Feb. 2018, pp.5618–5625. DOI: 10.1609/aaai.v32i1.12001.

[38] Liu D, Guo Q, Li W B, Lv J C. A multi-modal Chinese poetry generation model. In *Proc. the 2018 International Joint Conference on Neural Networks*, July 2018. DOI: 10.1109/IJCNN.2018.8489579.

[39] Liu Y S, Liu D, Lv J C. Deep poetry: A Chinese classical poetry generation system. In *Proc. the 34th AAAI Conference on Artificial Intelligence*, Feb. 2020, pp.13626–13627. DOI: 10.1609/aaai.v34i09.7100.

[40] Ghazvininejad M, Shi X, Choi Y J, Knight K. Generating topical poetry. In *Proc. the 2016 Conference on Empirical Methods in Natural Language Processing*, Nov. 2016, pp.1183–1191. DOI: 10.18653/v1/D16-1126.

[41] Greene E, Bodrumlu T, Knight K. Automatic analysis of rhythmic poetry with applications to generation and translation. In *Proc. the 2010 Conference on Empirical Methods in Natural Language Processing*, Oct. 2010, pp.524–533. DOI: 10.5555/1870658.1870709.

[42] Kolesnikov A, Lampert C H. PixelCNN models with auxiliary variables for natural image modeling. arXiv: 1612.08185, 2016. https://arxiv.org/abs/1612.08185, Sept. 2023.

[43] Gu J X, Cai J F, Wang G, Chen T. Stack-captioning: Coarse-to-fine learning for image captioning. In *Proc. the 32nd AAAI Conference on Artificial Intelligence*, Feb. 2018, pp.6837–6844. DOI: 10.1609/aaai.v32i1.12266.

[44] Guu K, Hashimoto T B, Oren Y, Liang P. Generating sentences by editing prototypes. *Trans. Association for Computational Linguistics*, 2018, 6: 437–450. DOI: 10.1162/tacl_a_00030.

[45] Pandey G, Contractor D, Kumar V, Joshi S. Exemplar encoder-decoder for neural conversation generation. In *Proc. the 56th Annual Meeting of the Association for Computational Linguistics*, July 2018, pp.1329–1338. DOI: 10.18653/v1/P18-1123.

[46] Wang Q Y, Zhou Z H, Huang L F, Whitehead S, Zhang B L, Ji H, Knight K. Paper abstract writing through editing mechanism. In *Proc. the 56th Annual Meeting of the Association for Computational Linguistics*, July 2018, pp.260–265. DOI: 10.18653/v1/P18-2042.

[47] Xia Y C, Tian F, Wu L J, Lin J X, Qin T, Yu N H, Liu T Y. Deliberation networks: Sequence generation beyond one-pass decoding. In *Proc. the 31st International Conference on Neural Information Processing Systems*, Dec. 2017, pp.1782–1792. DOI: 10.5555/3294771.3294941.

[48] Zhang H Y, Cai J J, Xu J J, Wang J. Pretraining-based

natural language generation for text summarization. In *Proc. the 23rd Conference on Computational Natural Language Learning*, Nov. 2019, pp.789–797. DOI: 10.18653/v1/K19-1074.

[49] Yan R. i, Poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proc. the 25th International Joint Conference on Artificial Intelligence*, July 2016, pp.2238–2244. DOI: 10.5555/3060832.3060934.

[50] Shen L, Guo X Y, Chen M. Compose like humans: Jointly improving the coherence and novelty for modern Chinese poetry generation. In *Proc. the 2020 International Joint Conference on Neural Networks*, July 2020. DOI: 10.1109/IJCNN48605.2020.9206888.

[51] Deng L M, Wang J, Liang H M, Chen H, Xie Z Q, Zhuang B J, Wang S J, Xiao J. An iterative polishing framework based on quality aware masked language model for Chinese poetry generation. In *Proc. the 34th AAAI Conference on Artificial Intelligence*, Feb. 2020, pp.7643–7650. DOI: 10.1609/aaai.v34i05.6265.

[52] Shao Y Z, Shao T, Wang M H, Wang P, Gao J. A sentiment and style controllable approach for Chinese poetry generation. In *Proc. the 30th ACM International Conference on Information & Knowledge Management*, Oct. 2021, pp.4784–4788. DOI: 10.1145/3459637.3481964.

[53] Ormazabal A, Artetxe M, Agirrezabal M, Soroa A, Agirre E. PoeLM: A meter- and rhyme-controllable language model for unsupervised poetry generation. In *Proc. the Findings of the Association for Computational Linguistics*, Dec. 2022, pp.3655–3670. DOI: 10.18653/v1/2022.findings-emnlp.268.

[54] Cho K, van Merriënboer B, Gulcehre C, Bahdanau D, Bougares F, Schwenk H, Bengio Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proc. the 2014 Conference on Empirical Methods in Natural Language Processing*, Oct. 2014, pp.1724–1734. DOI: 10.3115/v1/D14-1179.

[55] Mikolov T, Chen K, Corrado G, Dean J. Efficient estimation of word representations in vector space. arXiv: 1301.3781, 2013. https://arxiv.org/abs/1301.3781, Sept. 2023.

[56] Zhao H K, Cheng Y H, Zhang X, Zhu H S, Liu Q, Xiong H, Zhang W. What is market talking about? Market-oriented prospect analysis for entrepreneur fundraising. *IEEE Trans. Knowledge and Data Engineering*, 2023, 35(6): 6489–6503. DOI: 10.1109/TKDE.2022.3174336.

[57] Papineni K, Roukos S, Ward T, Zhu W J. BLEU: A method for automatic evaluation of machine translation. In *Proc. the 40th Annual Meeting on Association for Computational Linguistics*, July 2002, pp.311–318. DOI: 10.3115/1073083.1073135.

Ming He received his Ph.D. degree in computer science from University of Science and Technology of China, Hefei, in 2018. He is currently an advisory researcher with the AI Laboratory, Lenovo Research, Beijing. He was a visiting scholar at University of North Carolina at Charlotte in 2016–2017, was an algorithm researcher of DiDi, TAL (Tomorrow Advancing Life) in 2019–2021, and finished his post-doctor program at Department of Electronic Engineering of Shanghai Jiao Tong University, Shanghai, in 2022. His current research interests include recommendation system and reinforcement learning. He has published more than 15 papers in refereed journals and conference proceedings, e.g., TIP and SIGIR. He received the KSEM 2018 Best Research Paper Award and submitted more than 120 patents to Chinese National Patent Office.

Yan Chen received her M.S. degree in computer science from University of Science and Technology of China, Hefei, in 2018. She is currently a senior engineer with Baidu Incorporated, Beijing. Her current research interests include model robust, recommendation system and question answering system. She has published more than five papers in refereed journals and conference proceedings, e.g., EMNLP, DASFAA, and KSEM. She received the KSEM 2018 Best Research Paper Award.

Hong-Ke Zhao received his Ph.D. degree in computer science from University of Science and Technology of China, Hefei, in 2019. He is currently an associate professor with the College of Management and Economics, Tianjin University, Tianjin. His research interests include data mining, data-driven management, and knowledge and behavior computing. He has published more than 70 papers in refereed journals and conference proceedings, such as INFORMS Journal on Computing, TKDE, TBD, Industrial Marketing Management, Scientometrics, KDD, WWW, IJCAI and AAAI. He was the recipient of Distinguished Dissertation Award Nomination of CAAI (2019), and the Best Student Paper Award of CCML 2019. He has served regularly on the program committees of conferences, including KDD, AAAI, IJCAI, and WWW.

Qi Liu received his Ph.D. degree in computer science from University of Science and Technology of China, Hefei, in 2013. He is currently a professor with the School of Computer Science and Technology, University of Science and Technology of China, Hefei. His research interests include data mining, machine learning, and recommender systems. He was the recipient of KDD 2018 Best Student Paper Award and ICDM 2011 Best Research Paper Award. He was also the recipient of China Outstanding Youth Science Foundation in 2019. He is a member of IEEE.

Le Wu received her Ph.D. degree in computer science from University of Science and Technology of China, Hefei, in 2015. She is an associate professor with the School of Computer Science and Information Engineering, Hefei University of Technology, Hefei. Her general area of research is data mining, recommendation system, and social network analysis. She has published several papers in referred journals and conferences, such as the IEEE Transactions on Knowledge and Data Engineering, the ACM Transactions on Intelligent Systems and Technology, AAAI, IJCAI, KDD, SDM, and ICDM. She is the recipient of the Best of SDM 2015 Award. She is a member of IEEE.

Yu Cui received her M.E. degree in business administration from Beijing Institute of Technology, Beijing, in 2021. She is currently an algorithm engineer with the Risk Control Department, Opay, Beijing. She was an NLP algorithm engineer of TAL (Tomorrow Advancing Life) from 2018 to 2021. Her current research interests include text mining, dialogue systems, sentiment analysis, content risk control, and machine learning.

Gui-Hua Zeng received his Ph.D. degree in optics from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, in 1997. He is currently a professor with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai. His research interests include quantum cryptography, quantum optical communication, and quantum machine learning. He has published over 200 papers in refereed conferences and journals, e.g., IEEE Trans., Phys. Rev. A, and NPJ Quantum Information.

Gui-Quan Liu received his M.S. and Ph.D. degrees in computer science from University of Science and Technology of China, Hefei, in 1996 and 1999, respectively. He is currently an associate professor with the School of Computer Science and Technology, University of Science and Technology of China, Hefei. His main research interests include machine learning, data mining, social network analysis, and pattern recommendation. He has published over 60 papers in refereed conferences and journals.