Generating Thematic Chinese Poetry using Conditional Variational Autoencoders with Hybrid Decoders

Xiaopeng Yang, Xiaowen Lin, Shunda Suo, and Ming Li
David R. Cheriton School of Computer Science
Faculty of Mathematics, University of Waterloo
Waterloo, ON, Canada N2L 3G1
{x335yang, x65lin, sdsuo, mli}@uwaterloo.ca

Abstract

Computer poetry generation is our first step towards computer writing. Writing must have a theme. The current approaches of using sequence-to-sequence models with attention often produce non-thematic poems. We present a novel conditional variational autoencoder with a hybrid decoder adding the deconvolutional neural networks to the general recurrent neural networks to fully learn topic information via latent variables. This approach significantly improves the relevance of the generated poems by representing each line of the poem not only in a context-sensitive manner but also in a holistic way that is highly related to the given keyword and the learned topic. A proposed augmented word2vec model further improves the rhythm and symmetry. Tests show that the generated poems by our approach are mostly satisfying with regulated rules and consistent themes, and 73.42% of them receive an *Overall* score no less than 3 (the highest score is 5).

1 Introduction

Poetry is a beauty of simplicity. Its abstractness, concise formats, and rules provide regularities as the first target of language generation. Such regularity is especially amplified in the classical Chinese poetry, for example, the quatrains where each poem (1) consists of four lines, each with five or seven characters, (2) the last character in the second and fourth line follow the same rhythm, and (3) tonal pattern requests characters in particular positions hold particular tones in terms of Ping (level tone) and Ze (downward tone) [Wang, 2002]. An example of a quatrain written by Bo Wang, a famous poet in the Tang Dynasty, is shown in Table 1. As illustrated in Table 1, a good quatrain should follow all the three pattern regularities mentioned above.

Besides the rules, a poem is an expression of a certain theme or human emotion. It has to hold consistent semantic meanings and emotional expressions. It is not trivial to create a quatrain by certain rules of rhythm and tone, and express a consistent theme or some consistent feelings even by people. Automatically generating poetry that contains what we want it to express is a primary task of language generation.

Table 1: An example of five-character quatrain. The tonal pattern is shown at the end of each line, where 'P' indicates a level tone, 'Z' indicates a downward tone, and '*' indicates the tone can be either.

长江悲已滞，(*PPZZ)
Long stay by the Yangtze River,
万里念将归。(*ZZPP)
Thousands of Miles away from home,
况属高秋晚，(*ZPPZ)
Yellow Leaves in late autumn wind,
山中黄叶飞。(PPZZP)
Fall and float in hills, make me sad.

Major progress has been made in poetry generation [He *et al.*, 2012; Bahdanau *et al.*, 2014; Wang *et al.*, 2016a; Zhang *et al.*, 2017]. Even though the existing approaches have shown their great power in poetry automatic generation, they still suffer from a major problem: lack of consistent theme representation and unique emotional expression. Taking poem shown in Table 1 for instance, the consistent theme of this poem is nostalgia. Apparently, every single line of this poem is related closely to the theme and emotion. Recent work [Wang *et al.*, 2016b; Hopkins and Kiela, 2017] have tried to generate poems with the smooth and consistent theme by using topic planning scheme or similar word extensions. It is still hard for these methods to represent topics and use them to further improve the quality of generated poems.

In this paper, we try to solve the difficulty in learning the themes of poetry, meanwhile leveraging them to boost the generation of corresponding poems. As Variational AutoEncoders (VAE) [Kingma and Welling, 2013] have been proved effective in topic representation using learned latent variables for text generation [Bowman *et al.*, 2016; Serban *et al.*, 2017], we regard VAE as a possible solution. Moreover, since most written poems are composed under certain "intent," we seek for Conditional Variational AutoEncoders (CVAE), a recent modification of VAE, to generate diverse images/texts conditioned on certain attributes [Yan *et al.*, 2016b; Sohn *et al.*, 2015; Zhao *et al.*, 2017]. In our work, we hypothesize a part of the "intent" can be represented in the form of keywords as the conditions for VAE, and the other part can be expressed by the latent variables learned from CVAE. The general CVAE where both the encoder and decoder are RNNs usually faces the *vanishing la-*

tent variable problem [Bowman *et al.*, 2016] when applied directly to natural language generation. Thus, we present a novel CVAE with a hybrid decoder (CVAE-HD), which contains both deConvolutional Neural Networks (deCNN) and Recurrent Neural Networks (RNN), to fully learn information from the learned latent variables. In addition, we propose to add vertical slices of poems as additional sentences in training data for the word2vec model in order to further improve the rhythm and symmetry delivered in poems, and name this as an Augmented Word2Vec model (AW2V). We also propose a straightforward and easily applied automatically evaluation metric Rhythm Score Evaluation (RSE) to measure the poetry rule-consistency. Specifically, the contributions of this paper can be summarized as follows:

- We propose to use conditional variational autoencoders to learn the theme information from poetry lines. To the best of our knowledge, this represents the first attempt at using CVAE for poetry generation.
- We present a novel conditional variational autoencoder with a hybrid decoder combining deCNN with the general RNN, which demonstrates the capability of learning topic information from poems and also addressing the vanishing latent variable problem.
- We introduce an augmented word2vec model to improve the rhythm and symmetry delivered in poems. Experiments show that AW2V is not only able to boost the rule-consistency of generated poems, but also can be used to search characters representing similar semantic meanings in Chinese poems.
- We build a Chinese poetry generation system which can take users' writing intent into the generation process. The experimental results show that our system can generate good quatrains which satisfy the rules and have a consistent topic.

2 Related Work

Poetry generation is our first step toward experimenting language generation. According to the methodology used in different approaches, we categorize those methods into three major directions, i.e., approaches using rules/templates, approaches using Statistical Machine Translation (SMT) models and approaches using neural networks.

The first kind of approach is based on rules and/or templates, such as phrase search [Wu *et al.*, 2009] and genetic search [Zhou *et al.*, 2010].

The second kind of approach involves various statistical machine translation methods. Rather than designing algorithms to identify useful rules, the approaches using SMT models, whose parameters are derived from the analysis of bilingual text corpora, regard the previous line of each poem as the source language in the Machine Translation (MT) task and the posterior line as the target language sentence [Jiang and Zhou, 2008; He *et al.*, 2012].

Due to the fact that all the approaches mentioned above are based on the superficial meanings of words or characters, they suffer from the lack of deep understanding of the poems' semantic meaning. To address this issue, many approaches using neural networks have been proposed and attracted much attention in recent years. For example, [Zhang and Lapata, 2014] proposed an approach using Recurrent Neural Networks (RNN) that generate each new poem line character-by-character (see also [Hopkins and Kiela, 2017]), with all the lines generated previously as a contextual input. Experimental results show that quatrains of reasonable quality can be generated using this approach. Following this RNN-based approach, [Wang *et al.*, 2016a] proposed a character-based RNN treating a poem as an entire character sequence, which can be easily extended to various genres such as Song Iambics. This approach has the advantage of the flexibility and easy implementation, but the long-sequence generation process causes the instability of poetry theme. To avoid this situation, [Wang *et al.*, 2016a] further brought forward the attention mechanism [Bahdanau *et al.*, 2014] into the RNN-based framework, and encoded human intention to guide the poetry generation. [Yan, 2016] proposed an RNN-based poetry generation model with an iterative polishing scheme. Specifically, they encoded users' writing intent first and then decoded it using a hierarchical recurrent neural network. Recently, [Zhang *et al.*, 2017] proposed a memory-augmented neural model trying to imitate poetry writing process. This approach uses the augmented memory to refine poems generated via the neural model, which can balance the requirements of linguistic accordance and aesthetic innovation to some extent. Parallel efforts have been made in generating English poems. For instance, [Hopkins and Kiela, 2017] considered adding a list of similar words to a key theme.

We follow the third type of approach to automatically generate Chinese poetry. As introduced above, all the mentioned neural models attempt to produce poems with regulated rules, a consistent theme, and meaningful semantics, but none of them consider to represent poem theme and use it to further boost the results. To address this issue, we propose to use a novel conditional variational autoencoder with a hybrid decoder, in which the learned latent variables combined with conditional keywords are able to convey topic information of the entire poem.

3 Approaches

3.1 Overview

As most human poets write poems according to a sketch of ideas, we use a two-stage Chinese poem generation approach, i.e., writing intent representation and thematic poem generation. Specifically, our system can take a word, a sentence or even a document as input containing users' writing intent, and then generate rule-complied and theme-consistent poem sequentially using an improved conditional variational autoencoder. Similar work has been done in [Wang *et al.*, 2016b], the main distinction from our work to theirs is the implemented neural model is the conditional variational autoencoders in our work.

The framework of our Chinese poetry generation approach using the proposed Conditional Variational AutoEncoder with a Hybrid Decoder (CVAE-HD) is illustrated in Fig.1. Suppose an input query "冬天雪花纷飞" ("The snowflakes are flying in winter") is given, in the writing intent represen-

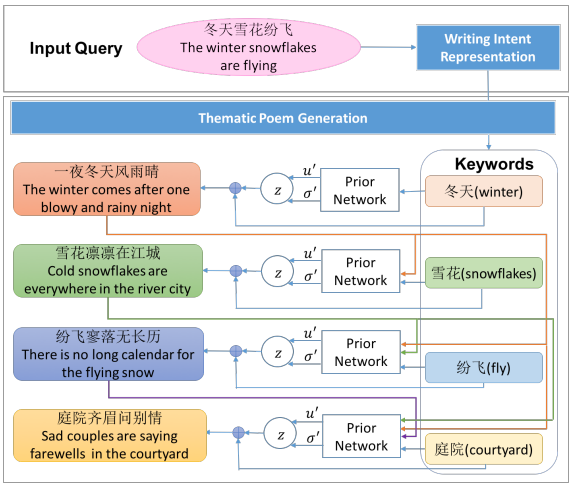

Figure 1: The framework of the proposed Chinese poetry generation approach. $\oplus$ denotes the concatenation of input vectors. [best viewed in color].

tation stage, the sentence is transformed into four keywords k_i ($i = 1, 2, 3, 4$), i.e., "冬天" (winter), "雪花" (snowflake), "纷飞" (fly), and "庭院" (courtyard), where k_i represents the sub-topic for the corresponding ith line l_i. In thematic poem generation stage, assuming that keywords are not enough to convey topic information for the entire poem, each line l_i is first encoded into a latent variable z_i to learn a distribution over potential writing intent by a prior network, and then generated by decoding from a concatenation of the learned latent variable z_i and the extracted or expanded keyword k_i. As a result, the poem is created automatically not only by the sub-topic provided by the corresponding keyword, but also the topic messages stored in latent variables, which are learned from the current line l_i, the previously generated lines $l_{1:(i-1)}$, and the corresponding keyword k_i. Note that the seven-character quatrain given in Fig.1 is produced automatically from our generation system.

3.2 Writing Intent Representation

Due to the fact that each line of a quatrain consists of five or seven characters, we hypothesize that the sub-topic of each line can be represented by one keyword. Therefore, it is important to evaluate the importance of words extracted from the input query provided by users. We use TextRank [Kingma and Ba, 2014] to measure the importance of different words.

In the graph of TextRank, a vertex represents each candidate word and edges between two words indicate their co-occurrence, where the edge weight is set according to the total count of co-occurrence strength between these two words. The TextRank score $S(V_i)$ is computed iteratively until convergence according to the following equation:

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

where w_{ji} is the weight of the edge between node V_j and V_i, $E(V_i)$ is the set of vertices connected with V_i, and d is a damping factor. Empirically, the damping factor d is usually set to 0.85, and the initial score of $S(V_i)$ is set to 1. When the number of extracted keywords from users' input query is less than the required one, we conduct the keyword extension in which the candidate word with the highest TextRank score is selected as the new keyword.

3.3 Conditional Variational Autoencoders with Hybrid Decoders

For Chinese poetry generation, since most human poets create poems based on a plain outline, we believe that keywords k_i ($i = 1, 2, 3, 4$) obtained from the first stage of our generation framework can partially represent users' writing intent, and regard them as the conditions c for CVAE.

We define the conditional distribution as $p(x, z|c) = p(x|z, c)p(z|c)$, and set the learning target to approximate $p(z|c)$ and $p(x|z, c)$ via deep neural networks parameterized by θ. CVAE is trained to maximize the conditional log likelihood of x given c, meanwhile minimizing the KL regularizer between the posterior distribution $p(z|x, c)$ and a prior distribution $p(z|c)$. We use a recognition network $q_\phi(z|x, c)$ and a prior network $p_\theta(z|c)$ to approximate the true posterior distribution $p(z|x, c)$ and the prior distribution $p(z|c)$. To sum up, the objective of the traditional CVAE takes the following form:

$$\begin{aligned} L(\theta, \phi; x, c)_{cvae} &= \mathbf{E} q_\phi(z|x,c)[log p_\theta(x|z,c)] \\ &\quad - KL(q_\phi(z|x,c)) \,||\, p_\theta(z|c)) \\ &\leq log p(x|c) \end{aligned} \quad (2)$$

As shown in Eqn.2, the generative process of x can be summarized as sampling a latent variable z from $p_\theta(z|c)$ and then generating x by $p_\theta(x|z, c)$. CVAE can be efficiently trained with the Stochastic Gradient Variational Bayes (SGVB) framework [Kingma and Ba, 2014] by maximizing the variational lower bound of the conditional log likelihood [Sohn *et al.*, 2015]. Fig.2 illustrates the training procedure of our proposed Conditional Variational Autoencoders with Hybrid Decoders (CVAE-HD). As shown in Fig.2, we use a Bidirectional Recurrent Neural Network (BRNN) [Kingma and Ba, 2014] with a Long Short Term Memory (LSTM) [Hochreiter and Schmidhuber, 1997] as an encoder to encode each concatenation of the current line l_i, the corresponding keyword k_i, and previously generated lines $l_{1:(i-1)}$ into fixed-size vectors by concatenating the last hidden states of the forward and backward RNN $h_i = \left[\overrightarrow{h_i}, \overleftarrow{h_i}\right]$. Then, x can be simply represented by h. We adopt multiple layers and residual connections [He *et al.*, 2016] between layers to learn a describable h . We suppose z follows a multivariate Gaussian distribution with a diagonal covariance matrix, thus the recognition network $q_\phi(z|x, c) \sim \mathcal{N}(\mu, \sigma^2\mathbf{I})$ and the prior network $p_\theta(z|c) \sim \mathcal{N}(\mu', \sigma'^2\mathbf{I})$, and then we have:

$$\begin{bmatrix} \mu \\ log(\sigma^2) \end{bmatrix} = W_r \begin{bmatrix} x \\ c \end{bmatrix} + b_r \quad (3)$$

$$\begin{bmatrix} \mu' \\ log(\sigma'^2) \end{bmatrix} = \mathbf{MLP}_p(c) \quad (4)$$

We use a reparametrization trick [Kingma and Welling, 2013] to sample z from the recognition network $\mathcal{N}(\mu, \sigma^2\mathbf{I})$ during

training and $\mathcal{N}(\mu', \sigma'^2\mathbf{I})$ predicted by the prior network during testing. The initial state $s_0 = W_d[z, c] + b_d$ is used for a RNN decoder.

Since it is easy for CVAE to ignore the latent variable z when directly using an RNN decoder, inspired by [Semeniuta *et al.*, 2017], we propose to use a hybrid decoder in CVAE as shown in Fig.2 and name the novel CVAE as Conditional Variational Autoencoders with Hybrid Decoders (CVAE-HD). The hybrid decoder is composed of deConvolutional Neural Networks (deCNN) [Radford *et al.*, 2015; Gulrajani *et al.*, 2016] and recurrent neural networks. The reason we introduce deCNN as a part of the decoder in CVAE is to build the connection of each element in x with the learned latent variable z. Then, the probability of the generated sequence x can be represented as $P(x_1, ..., x_n|z, c) = \prod_i P(x_i|z, c)$. However, it is hard for a fully feed-forward architecture to learn the sequential information between the element in x. A multi-layer LSTM decoder similar with the encoder is added on top of deCNN layers to model $P(x_1, ..., x_n|z, c) = \prod_i P(x_i|x_{i-1}, ..., x_1, z, c)$.

3.4 Optimization

Although CVAE has achieved impressive results in image generation, it is non-trivial to adapt it to natural language generators due to the *vanishing latent variable problem*. KL annealing [Bowman *et al.*, 2016] gradually increasing the weight of the KL term from 0 to 1 during training plays a powerful role in dealing with this problem. Another solution word drop decoding, which sets a certain percentage of the target words to 0, may hurt the performance when the drop rate is too high. Thus, we adopt KL annealing instead of word drop decoding during training for CVAE.

Beyond that, we propose an auxiliary solution to help further solve the above problem, i.e., we add an additional deCNN reconstruction loss term to Eqn.2 and regularize it with a weighting parameter α. Therefore, the loss function of our proposed CVAE-HD can be represented as below:

$$L_{cvae-hd} = L_{cvae} + \alpha L_{dcnn} \quad (5)$$

in which the second term is computed from the activations of the last deconvolutional layer $L(\theta, \phi; x, c)_{dcnn} = \mathbf{E}q_\phi(z|x, c)[logp_\theta(x|z, c)]$.

Since we use the combination of both keywords extracted from users' query and the latent variable z learned from CVAE to represent poetry theme, the representation of keywords is the key for the performance to some extent. Therefore, we try to mine the nature of quatrains to obtain a good representation of poetry word. We notice that for some lines in quatrains, mostly the third and the fourth line, corresponding characters from the same position in these two lines often match each other by certain constraints on semantic and/or syntactic relatedness.

Taking two lines "千山鸟飞绝 (A thousand mountains without birds flying), 万径人踪灭 (Ten thousand paths without a footprint)" of the famous five-quatrain "江雪" (River Snow) as an example, the characters "千" (thousand) and "万" (ten thousand) both represent numbers, meanwhile "绝" (gone) and "灭" (disappeared) both deliver similar meanings of nonexistence. Even though the constraints

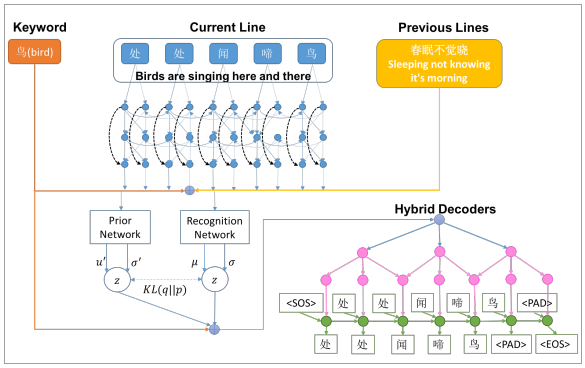

Figure 2: The training procedure of the proposed Conditional Variational Autoencoders with Hybrid Decoders. The black dashed lines represent the residual connection between layers [best viewed in color].

of quatrains are not as strict as the Chinese antithetical couplets [Yan *et al.*, 2016a], we propose to initialize the word-embedding vectors using an Augmented Word2Vec model (AW2V) to further enhance the rhythm and symmetry delivered in poems. This model adds vertical slices of poems as additional sentences to the training data based on the word2vec [Mikolov *et al.*, 2013]. AW2V is not only able to boost the optimization of CVAE and improve the rule-consistency of generated poems, but also can be used to search characters representing similar semantic meanings in Chinese poems.

4 Experimental Setup

4.1 Dataset

Two large-scale datasets are used in our experiments. The first dataset is a Chinese poem corpus (CPC) containing 284,899 traditional Chinese poems in various genres, including Tang quatrains, Song Iambics, Yuan Songs, and Ming and Qing poems. We use this dataset to train the word-embedding for Chinese characters. Since we focus on generating quatrains which have four lines with the same length of five or seven characters in each line, we filter 76,305 quatrains from CPC, named as Chinese quatrain corpus (CQC), to train the neural network model. Specifically, we randomly choose 2,000 poems for validation, 1,000 poems for testing, and other non-overlap ones for training. We segment all the poems into words and calculate the TextRank score for each word. Then, the word with the highest TextRank score is selected as the keyword for each line so that each quatrain owns four keywords.

4.2 Training

We choose the 6,000 most frequently used characters as the vocabulary. The dimension of word-embedding vectors is set to 128. The recurrent hidden layers of the encoder and the RNN part of the hybrid decoder contain 128 hidden units, and the number of layers is both set to 4. We use 3 deconvolutional layers with the ReLU non-linearity in the deCNN part of the hybrid decoder. The kernel size is set to 3 and the stride is 2. The number of feature maps is [256, 128, 64] for each layer respectively. The weighting parameter α is set

Table 2: The language modeling results and the performance on automatic metrics on the test dataset. We report negative log likelihood (NLL) and perplexity (PPL), and the KL component of NLL is given in parentheses. Note that the reported BLEU scores are normalized to [0, 1]. The mean score of the proposed Rhythm Score Evaluation (RSE) is reported in the last column. The corresponding best scores are shown in bold.

Approach	PPL	NNL(KL)	BLEU-1	BLEU-2	BLEU-3	BLEU-4	RES
Ground truth	-	-	-	-	-	-	0.8975
AS2S	55.5255	4.0168(-)	0.4296	0.3596	0.3045	0.2640	0.5046
AS2S+AW2V	54.1752	3.9922(-)	0.4319	0.3625	0.3076	0.2669	0.5076
CVAE	52.1154	3.9535(0.0083)	0.4366	0.3605	0.3046	0.2637	0.5137
CVAE+AW2V	52.0516	3.9522(0.0126)	0.4382	0.3630	0.3073	0.2663	0.5149
CVAE-HD+AW2V	**51.7236**	**3.9459(0.0163)**	**0.4405**	**0.3644**	**0.3084**	**0.2672**	**0.5245**

to 0.6. We use 64-dimensional latent variables. Parameters of our model were randomly initialized over a uniform distribution with support [-0.08,0.08]. The model is trained using the AdaDelta algorithm [Zeiler, 2012], where the mini-batch is set to 64 and the learning rate is 0.001. The dropout technique [Srivastava *et al.*, 2014] is also adopted and the dropout rate is set to 0.2. The perplexity value on the validation set is used for the early stop of training to avoid an overfitting learned model.

5 Evaluation

Generally, it is difficult to judge the quality of poems generated by computers. We conduct both automatic and human evaluation to verify the feasibility and availability of our proposed Chinese poetry generation approach.

For the comparative approach, we mainly compare our proposed approach with the attention-based sequence-to-sequence model (AS2S) presented in [Wang *et al.*, 2016b], which has been proved to be capable of generating different genres of Chinese poems. The reasons we choose AS2S to compare rather than others can be summarized into two aspects. First, this model has been fully compared with other previous methods such as SMT, RNNLM, RNNPG, and ANMT, and proved better than all of them. Second, the first generation phase of our proposed approach, i.e., writing intent representation, is similar to the procedure introduced in [Wang *et al.*, 2016b] while the second phase is completely different. Therefore, through comparing our framework with theirs, we can inspect the effects of our proposed Conditional Variational AutoEncoders with Hybrid Decoders (CVAE-HD).

5.1 Automatic Evaluation

Poetry Modeling Results

The language modeling results on the test dataset of CQC are shown in Table 2, in which the reconstruction perplexity (PPL), negative log likelihood (NLL) and the KL component (KL) are reported. In addition to this, BLEU evaluation method [Papineni *et al.*, 2002], which is famous for the evaluation the quality of the text, is also reported in Table 2. We use BLEU-1 to 4, and normalize them to $[0, 1]$ scale.

Since Chinese quatrains have strict regulations and should follow particular tonal and structural rules, we propose a new Rhythm Score Evaluation (RSE) to automatically measure the rule-consistency of poems. We define RSE as

$$Rhy(l) = \begin{cases} 0, & cnt(l) \notin \{5, 7\} \\ 0.5, & rule(l) \in T \ or \ R \\ 1, & rule(l) \in T \ and \ R \end{cases} \quad (6)$$

where l represents each line of poems, $cnt(l)$ is the number of characters of l, $rule(l)$ is the rule of l, and T and R represent the set of tonal patterns and rhyming patterns severally. The results of mean score in terms of RSE are demonstrated in the last column of Table 2. A higher mean score indicates approaches owning better capability of generating poems with regulated rhythm and structure. Note that the ground truth represents the humanly written poems in the test dataset.

From Table 2, we can find that, compared with AS2S as the baseline, both CVAE+AW2V and CVAE-HD+AW2V outperform in terms of all the metrics. Note that we represent the approach using our proposed augmented word2vec model (AW2V) by appending a plus sign to the original method, e.g., AS2S+AW2V. Compared with AS2S+AW2V and AS2S, and CVAE+AW2V and CVAE, the improvement by adding AW2V can be found in both AS2S and CVAE, which demonstrates the advantage of AW2V in the optimization for poetry modeling. Beyond this, CVAE-HD, the proposed novel CVAE with a hybrid decoder, outperforms CVAE especially in the terms of KL. This proves that the hybrid decoder can relieve the pressure of vanishing latent variables to a certain extent. We notice that due to the simplification of poetry, although KL annealing and our proposed hybrid decoder are all adopted during training, it is harder for poetry generation to tackle the vanishing latent variable problem than general natural language generation.

Poetry Character Similarity

We measure the similarity of poetry characters to verify the superiority of our proposed AW2V model over the original word2vec (W2V) one.

Taking the poetry "江雪" (River Snow) mentioned in Section 3.4 as an instance, the similarity between 千 (thousand) and 万 (ten thousand) using AW2V is 0.4389, while 0.4039 using W2V. It is worth noticing that 绝 (gone) and 灭 (disappeared) get a 0.2745 similarity score in AW2V model, while only get 0.0205 in W2V. Beyond that, we can use AW2V to search similar words. For instance, if we search similar words for 年 (year), we obtain 旬 (ten days), 番 (multiple times), and 时 (time) which are all time-related Chinese characters.

Table 3: The five/seven-character generated quatrain based on the given query “蜡烛 (candle)”/“梅 (wintersweet)”.

蜡烛今宵尽， Burned candle flickered at dawn, 残灯隔户人。 A dim light shone on the man home alone. 衣襟因酒别， Drinking, my body and mind flowed, 何况雪中人。 While you were fainting in the snow.	寒梅寂寞是无家， Lonely, a plum blossom is homeless, 未折惊心待岁华。 Apprehensive, the ephemeral beauty will wither. 折得一枝凝瘦骨， Hey, don’t worry. I will pick a twig in the woods, 砌间长笛有梅花。 And house it in my flute.

Figure 3: The website interface of our poetry generation system in which users are asked to input the query and rate the generated poem based on certain metrics [best viewed in color].

5.2 Human Evaluation

Online Test

We build a web-based environment for our poetry generation system whose interface is illustrated in Fig.3. Using this website, users can input any arbitrary query as the topic to generate a computer-written poem using our proposed CVAE-HD.

We invited users who are well-educated and have a great passion for poetry writing to participate our human evaluation. All the participants are asked to rate poems in the score range $[1, 5]$ based on five subjective evaluation metrics including *Readability* (if the sentences read smoothly and fluently), *Consistency* (if the entire poem delivers a consistent theme), *Aesthetic* (if the quatrain stimulates any aesthetic feeling), *Evocative* (if the quatrain expresses meaningful emotion), and *Overall* (if the quatrain is overall well written).

As shown in Fig.3, the illustrated example contains the user input query “秋风” (Autumn Wind) and the corresponding generated poem “秋风堪味著青华 (The lush growth of trees colors the early autumn wind)，松柏瑶林鹤夜来 (A group of cranes flies through the forest in the evening)。九月黄昏冰影里 (It is only September and the frost is shivering at dusk)，枝头遥隔旧香来 (The branches remind me of the distance separating us)。”. We notice that the generated poem not only contains the related words of “autumn wind,” e.g., “cranes” and “September,” but also delivers consistently gloomy theme and emotion. This poem, moreover, intuitively conforms to the tonal and structural rules of quatrains.

Up to now, we collect 139 quatrains based on all the random queries input by the human-evaluation participants. On average, we obtain a 3.43 *Readability* score, a 3.15 *Consistency* score, a 3.26 *Aesthetic* score, a 3.16 *Evocative* score, and a 3.22 *Overall* score. Among all the generated poems, 73.42% of them receive an *Overall* score no less than 3.

We also give some other specific examples based on various given queries. Two quatrains produced by our poetry generation system are shown in Table 3.

6 Conclusions

In this work, we have studied poetry generation. We present a two-step generation approach including writing intent representation and thematic poem generation to imitate the poem creation process by human poets. We have proposed a conditional variational autoencoder with a hybrid decoder to mine the implicit topic information contained within poems lines. An augmented word2vec model has also been proposed to further enhance the rhythm and symmetry delivered in poems and improve the training procedure. The generative neural model can incorporate more flexibility to represent the theme message by learning latent variables.

We conduct the experiments on several evaluation metrics and compare our proposed approach with some existing ones. Experimental results demonstrate that our proposed poetry generation approach can produce satisfying quatrains with regulated rules and consistent themes. Our proposed conditional variational autoencoder with a hybrid decoder has been proved to outperform the attention-based sequence-to-sequence model.

Currently, we are working on using reinforcement learning to further improve the poetry quality and generating different literature, such as lyrics and compositions.

Acknowledgments

This work is partially supported by China’s National Key Research and Development Program under grant 2016YFB1000902 and 2018YFB1003202, Beijing Advanced Innovation Center for Imaging Technology BAICIT-2016031, Ningbo Science and Technology Innovation team No. 2014B82014, Canada’s NSERC OGP0046506, and Canada Research Chair Program.

References

[Bahdanau *et al.*, 2014] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473*, 2014.

[Bowman *et al.*, 2016] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In *CoNLL*, 2016.

[Gulrajani *et al.*, 2016] Ishaan Gulrajani, Kundan Kumar, Faruk Ahmed, Adrien Ali Taiga, Francesco Visin, David Vazquez, and Aaron Courville. Pixelvae: A latent variable model for natural images. *arXiv preprint arXiv:1611.05013*, 2016.

[He *et al.*, 2012] Jing He, Ming Zhou, and Long Jiang. Generating chinese classical poems with statistical machine translation models. In *AAAI*, 2012.

[He *et al.*, 2016] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

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

[Hopkins and Kiela, 2017] Jack Hopkins and Douwe Kiela. Automatically generating rhythmic verse with neural networks. In *ACL(1)*, pages 168–178, 2017.

[Jiang and Zhou, 2008] Long Jiang and Ming Zhou. Generating chinese couplets using a statistical mt approach. In *ACL(1)*, pages 377–384, 2008.

[Kingma and Ba, 2014] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[Kingma and Welling, 2013] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

[Mikolov *et al.*, 2013] Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*, 2013.

[Papineni *et al.*, 2002] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *ACL*, pages 311–318, 2002.

[Radford *et al.*, 2015] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015.

[Semeniuta *et al.*, 2017] Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth. A hybrid convolutional variational autoencoder for text generation. In *EMNLP*, 2017.

[Serban *et al.*, 2017] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*, pages 3295–3301, 2017.

[Sohn *et al.*, 2015] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *NIPS*, pages 3483–3491, 2015.

[Srivastava *et al.*, 2014] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of machine learning research*, 15(1):1929–1958, 2014.

[Wang *et al.*, 2016a] Qixin Wang, Tianyi Luo, Dong Wang, and Chao Xing. Chinese song iambics generation with neural attention-based model. In *IJCAI*, pages 2943–2949, 2016.

[Wang *et al.*, 2016b] Zhe Wang, Wei He, Hua Wu, Haiyang Wu, Wei Li, Haifeng Wang, and Enhong Chen. Chinese poetry generation with planning based neural network. In *COLING*, 2016.

[Wang, 2002] Li Wang. *A Summary of Rhyming Constraints of Chinese Poems (Shi Ci Ge Lv Gai Yao)*, volume 1. Beijin Press., 2002.

[Wu *et al.*, 2009] Xiaofeng Wu, Naoko Tosa, and Ryohei Nakatsu. New hitch haiku: An interactive renku poem composition supporting tool applied for sightseeing navigation system. In *ICEC*, pages 191–196, 2009.

[Yan *et al.*, 2016a] Rui Yan, Cheng-Te Li, Xiaohua Hu, and Ming Zhang. Chinese couplet generation with neural network structures. In *ACL(1)*, pages 2347–2357, 2016.

[Yan *et al.*, 2016b] Xinchen Yan, Jimei Yang, Kihyuk Sohn, and Honglak Lee. Attribute2image: Conditional image generation from visual attributes. In *ECCV*, pages 776–791, 2016.

[Yan, 2016] Rui Yan. ipoet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *IJCAI*, pages 2238–2244, 2016.

[Zeiler, 2012] Matthew D Zeiler. Adadelta: an adaptive learning rate method. *arXiv preprint arXiv:1212.5701*, 2012.

[Zhang and Lapata, 2014] Xingxing Zhang and Mirella Lapata. Chinese poetry generation with recurrent neural networks. In *EMNLP*, pages 670–680, 2014.

[Zhang *et al.*, 2017] Jiyuan Zhang, Yang Feng, Dong Wang, Yang Wang, Andrew Abel, Shiyue Zhang, and Andi Zhang. Flexible and creative chinese poetry generation using neural memory. In *ACL*, 2017.

[Zhao *et al.*, 2017] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *ACL*, 2017.

[Zhou *et al.*, 2010] Cheng-Le Zhou, Wei You, and Xiaojun Ding. Genetic algorithm and its implementation of automatic generation of chinese songci. *Journal of Software*, 21(3):427–437, 2010.