Generating Chinese Couplets and Quatrain Using a Statistical Approach *

Ming Zhou[a], Long Jiang[a], and Jing He[b]

[a]Microsft Research Asia,
Sigma Centre, Haidian, Beijing, 100109, P.R. China
{mingzhou, longj}@microsoft.com
[b]Dept. of Computer Science, Tsinghua University
hejing2929@gmail.com

Abstract. We propose a novel statistical approach to automatically generate Chinese couplets and Chinese poetry. For Chinese couplets, the system takes as input the first sentence and generates as output an N-best list of second sentences using a phrase-based SMT model. A comprehensive evaluation using both human judgments and BLEU scores has been conducted and the results demonstrate that this approach is very successful. We then extended this approach to generate classic Chinese poetry using the quatrain as a case study. Given a few keywords describing a user's intention, a statistical model is used to generate the first sentence. Then a phrase-based SMT model is used to generate the other three quatrain sentences one by one. Evaluation using human judgment over individual lines as well as the quality of the generated poem as a whole demonstrates promising results.

Keywords: Chinese couplets, Chinese classic poetry, automatic generation.

1 Introduction

This paper presents a novel approach to generate Chinese couplets and classic Chinese poetry with a Statistical Machine Translation (SMT) model. Chinese antithetical couplets, called "duìlián", form a special type of poetry composed of two sentences. The two sentences making up a couplet are called the "first sentence" (FS) and the "second sentence" (SS) respectively. The quatrain and regulated verse are the two most important forms of classic Chinese poetry. A quatrain is composed of 4 lines and regulated verse is composed of 8 lines with additional rules governing structure. Each line of the quatrain or regulated verse is composed of 5 Chinese characters (五言) or 7 characters (七言).

For Chinese couplets, the general process of generating a SS given a FS is like this: for each word in the FS, find some words that can be used as the counterparts in the SS; then from the word lattice, select one word at each position in the SS so that the selected words form a fluent sentence satisfying the constraints of Chinese couplets. This process is similar to translating a source language sentence into a target language sentence without word insertion, deletion and reordering, but the target sentence should satisfy some linguistic constraints. Based on this observation, we propose a multi-phase statistical machine translation approach to generate the SS. First, a phrase-based statistical machine translation (SMT) model is applied to generate an

* This paper summarizes our work on automatic generation of Chinese couplets (Jiang & Zhou, 2008) and automatic generation of poetry (He et al., to appear), both were conducted in Natural Language Group at Microsoft Research Asia.

N-best list of SS candidates. Then, a set of filters based on linguistic constraints for Chinese couplets is used to remove low quality candidates. Finally, a Ranking SVM is applied to rerank the candidates. A comprehensive evaluation using both human judgments and BLEU scores has been conducted and the results demonstrate that this approach is very successful.

We extended this approach to generate classic Chinese poetry using the quatrain as a case study. Given a few keywords describing a user's intention, a statistical model is used to generate the first sentence. Then a phrase-based SMT model is used to generate the other three quatrain sentences one by one. Preliminary evaluation using human judgment over individual lines as well as the quality of the generated poem as a whole demonstrates promising results.

2 Linguistic Constraints of Couplets

We use an example of a Chinese couplet to explain the linguistic constraints of Chinese couplets in Table 1 with the correspondence between individual words of the FS and SS. It means that *the sea is wide so that fish jump at their pleasure, and the sky is high so that bird flies unrestrictedly*.

Table 1: An example of Chinese couplets

FS:	海(hai) sea	阔(kuo) wide	凭(pin) allow	鱼(yu) fish	跃(yue) Jump
	|	|	|	|	|
SS:	天(tian) sky	高(gao) high	任(ren) permit	鸟(niao) bird	飞(fei) fly

A couplet must conform to the following linguistic constraints:

1) The two sentences of a couplet agree in length and word segmentation.
2) In Chinese, every character is pronounced either level (平) or oblique (仄). The character at the end of the FS should be oblique (pronounced in a sharp downward tone); the character at the end of the SS should be level (pronounced in a level tone).
3) Corresponding words in the two sentences should agree in their part of speech and characteristics.
4) The contents of the two sentences should be related, but not duplicated.
5) The two sentences should be identical in their writing styles. For instance, if there is a repetition of words, characters, or pronunciations in the FS, the SS should contain an identical repetition.

Given a FS, writing a good SS is a difficult task because the SS must conform to constraints on syntax, rhyme and semantics, as described above. It requires the writer to innovatively use extensive knowledge in multiple disciplines.

3 SMT-Based Couplet Generation Model

In this paper, a multi-phase statistical machine translation (SMT) approach is designed, where an SMT system generates an N-best list of candidates and then a ranking model is used to determine the new ranking of the N-best results using additional features. This approach is similar to reranking approaches of SMT (Och and Ney, 2004). In our SMT system, a phrase-based log-linear model is applied where two phrase translation models, two lexical weights and a language model are used to score the output sentences, and a monotone phrase-based decoder is employed to get the N-best results. Then a set of filters based on linguistic constraints of Chinese couplets are used to remove low quality candidates. Finally a Ranking SVM model is used to rerank the candidates using additional features like word associations, etc.

3.1 Phrase-based SMT Model

Given a FS denoted as $F = \{f_1, f_2, ..., f_n\}$, our objective is to seek a SS denoted as $S = \{s_1, s_2, ..., s_n\}$, where f_i and s_i are Chinese characters, so that $p(S|F)$ is maximized. Following Och and Ney (2002) which departs from the traditional noisy-channel approach and uses a more general log-linear model, the S^* that maximizes $p(S|F)$ can be expressed as follows:

$$\begin{aligned} S^* &= \arg \max_S \; p(S \mid F) \\ &= \arg \max_S \; \sum_{i=1}^{M} \lambda_i \log \; h_i(S, F) \end{aligned} \tag{1}$$

where the $h_i(S,F)$ are feature functions and M is the number of feature functions. In our design, characters are used instead of words as translation units to form phrases. This is because Chinese couplets use dense language mostly following the similar style of ancient Chinese and most of words contain only one character. With this character based approach, we can avoid unexpected errors due to segmentation ambiguities and OOV (Out of Vocabulary) words.

Among features commonly used in phrase-based SMT, five features, listed in Table 2, were selected for our model. To apply phrase-based features, S and F are segmented into phrases $\overline{s_1}...\overline{s_I}$ and $\overline{f_1}...\overline{f_I}$, respectively. We assume a uniform distribution over all possible segmentations.

Table 2: Features in our SMT Model.

$h_1(S,F) = \prod_{i=1}^{I} p(\overline{f_i} \mid \overline{s_i})$	Phrase translation model
$h_2(S,F) = \prod_{i=1}^{I} p(\overline{s_i} \mid \overline{f_i})$	Inverted phrase translation model
$h_3(S,F) = \prod_{i=1}^{I} p_w(\overline{f_i} \mid \overline{s_i})$	Lexical weight
$h_4(S,F) = \prod_{i=1}^{I} p_w(\overline{s_i} \mid \overline{f_i})$	Inverted lexical weight
$h_5(S,F) = p(S)$	Language model

Phrase translation model (PTM)

In a phrase-based SMT model, phrases can be any substring that may not necessarily be linguistically motivated units. In our implementation, we extract phrases of up to 4-character-grams. In a Chinese couplet, there is generally a direct one-to-one mapping between words at same position in the FS and SS, respectively. As a result, the i[th] character/phrase in F is exactly "translated" into the i[th] character/phrase in S. Based on this rule, the phrase translation probability $p(\overline{f_i} \mid \overline{s_i})$ can be estimated by relative frequency in a training corpus:

$$p(\overline{f_i} \mid \overline{s_i}) = \frac{count(\overline{f_i}, \overline{s_i})}{\sum_{r=1}^{m} count(\overline{f_r}, \overline{s_i})} \tag{2}$$

where m is the number of distinct phrases that can be mapped to the phrase $\overline{s_i}$ and $count(\overline{f_i}, \overline{s_i})$ is the number of occurrences that $\overline{f_i}$ and $\overline{s_i}$ appear at the corresponding positions in a couplet.

The inverted phrase translation model $p(\overline{s_i} \mid \overline{f_i})$ has been proven useful in previous SMT research work (Och and Ney, 2002); so we also include it in our phrase-based SMT model.

Lexical weight (LW)

Previous research work on phrase-based SMT has found that it is important to validate the quality of a phrase translation pair (Koehn et al., 2003). A good way to do this is to check its lexical weight $p_w(\overline{f_i} \mid \overline{s_i})$, which indicates how well its words translate to each other:

$$p_w(\overline{f_i} \mid \overline{s_i}) = \prod_{j=1}^{Ni} p(f_j \mid s_j) \quad (3)$$

where Ni is the number of characters in $\overline{f_i}$ or $\overline{s_i}$, f_j and s_j are characters in $\overline{f_i}$ and $\overline{s_i}$ respectively, and $p(f_j \mid s_j)$ is the character translation probability of s_j into f_j. Like in phrase translation probability estimation, $p(f_j \mid s_j)$ can be computed by relative frequency:

$$p(f_j \mid s_j) = \frac{count(f_j, s_j)}{\sum_{r=1}^{m} count(f_r, s_j)} \quad (4)$$

where m is the number of distinct characters that can be mapped to the character s_j and $count(f_j, s_j)$ is the number of occurrences that s_j and f_j appear at the corresponding positions in a couplet.

Like the phrase translation model, we also use an inverted lexical weight $p_w(\overline{s_i} \mid \overline{f_i})$ in addition to the conventional lexical weight $p_w(\overline{f_i} \mid \overline{s_i})$ in our phrase-based SMT model.

Language model (LM)

A character-based trigram language model with Katz back-off is constructed from the training data to estimate the language model $p(S)$ using Maximum Likelihood Estimation.

3.2 Data Collection and Model Training

We used the method proposed by (Fan et al., 2007) to recursively mine those couplets with the help of some seed couplets. The method automatically learns patterns in a page containing Chinese couplets and then applies the learned patterns to extract more Chinese couplets. In addition, we found some online forums where Chinese couplet fans challenge each other. When a new FS is posted on the forums, many other people submit their SSs in response, each with different meaning and word usage. Using various web mining approaches, we finally collected 670,000 couplets. We also mined pairs of sentences of poetry which satisfied the constraints of couplets although they were not originally intended as couplets. For instance, in eight-line Tang poetry, the third and fourth sentences and the fifth and sixth sentences form pairs basically satisfying the constraints of Chinese couplets. Therefore, an additional 300,000 sentence pairs were obtained, yielding a total of 970,000 sentence pairs of training data.

Because the relationships between words and phrases in the FS and SS are usually reversible, to alleviate the data sparseness, we reverse the FS and SS in the training couplets and merge them with original training data for estimating translation probabilities. To smooth the language model, we add about 1,600,000 sentences which are not necessarily couplets, derived from ancient Chinese poetry, for language model training. To estimate the weights λ_i in formula (1), we use the Minimum Error Rate Training (MERT) algorithm, which is widely used for phrase-based SMT model training (Och, 2003). The training data and criteria (BLEU) for MERT will be explained in Subsection 4.1.

3.3 Linguistic Filters

To reflect the linguistic constrains, a set of filters is used to remove candidates that violate linguistic constraints. For instance, **Repetition filter** removes candidates based on various rules related to word or character repetition. One such rule requires that if there are multiple characters that are identical in the FS, then the corresponding characters in the SS should be identical too. Conversely, if there are no identical words in the FS, then the SS should have no

identical words. Similarly, **Pronunciation repetition filter, Character decomposition filter, Phonetic harmony filter** are used to remove the candidates violating the requirements on the pronunciation, character decomposition and phonetic harmony.

3.4 Reranking Based on Multiple Features

In many cases, long-distance constraints are very helpful in selecting good SSs, however, it is difficult to incorporate them in the framework of the dynamic programming decoding algorithm. To solve this issue, we designed an SVM-based reranking model incorporating long-distance features to select better candidates.

As shown in formula (5), $\vec{x}$ is the feature vector of a SS candidate, and $\vec{w}$ is the vector of weights. $\langle \cdot, \cdot \rangle$ stands for an inner product. f is the decision function with which we rank the candidates.

$$f_{\vec{w}}(\vec{x}) = \langle \vec{w}, \vec{x} \rangle \tag{5}$$

Besides the five features used in the phrase-based SMT model, additional features for reranking can be added into this framework such as the two as follows (Jiang & Zhou, 2008).

- Mutual information (MI) score:
- MI-based structural similarity (MISS) score:

4 Experimental Results

4.1 Evaluation Method

Automatic evaluation is very important for parameter estimation and system tuning. An automatic evaluation needs a standard answer data set and a metric to show for a given input sentence the closeness of the system output to the standard answers. Since generating the SS given the FS is viewed as a kind of machine translation process, the widely accepted automatic SMT evaluation methods may be applied to evaluate the generated SSs.

BLEU (Papineni et al., 2002) is widely used for automatic evaluation of machine translation systems. It measures the similarity between the MT system output and human-made reference translations. The BLEU metric ranges from 0 to 1 and a higher BLEU score indicates better translation quality.

$$BLEU = BP \bullet \exp(\sum_{n=1}^{N} w_n \log p_n) \tag{6}$$

Some adaptation is necessary to use BLEU for evaluation of our couplet generator. First, p_n, the n-gram precision, should be position-sensitive in the evaluation of SSs. Second, BP, the brevity penalty, should be removed, because all system outputs have the same length and it has no effect in evaluating SSs. Moreover, because the couplet sentences usually have fewer than 10 characters, we set n to 3 for the evaluation of SSs, while in MT evaluation n is often set to 4.

It is important to note that the more reference translations we have for a testing sentence, the more reasonable the evaluation score is. From couplet forums mentioned in Subsection 3.2, we collected 1,051 FSs with diverse styles and each of them has over 20 unique SS references. After hand removal of some noisy references, each of them has 24.3 references on average. The minimum and maximum number of references is 20 and 40. Out of these data, 600 were selected for MERT training and the remaining 451 for testing.

4.2 Feature Evaluation

We conducted some experiments incrementally to evaluate the features used in our phrase-based SMT model and reranking model. All testing data are used. The results are listed below.

Table 4: Feature Evaluation.

	Features	BLEU
Phrase-based SMT Model	Phrase TM(PTM) + LM	0.276
	+ Inverted PTM	0.282
	+ Lexical Weight (LW)	0.315
	+ Inverted LW	0.348
Ranking SVM	+ Mutual information (MI)	0.356
	+ MI-based structural similarity	0.361

As shown in Table 4, with two features: the phrase translation model and the language model, the phrase-based SMT model can achieve a 0.276 of BLEU score. When we add more features incrementally, the BLEU score is improved consistently.

4.3 Human Evaluation

In addition to BLEU evaluation, we also carried out human evaluation. We select 100 FSs from the log data of our couplet web service (http://duilian.msra.cn) which was launched in 2005. For each FS, 10 best SS candidates are generated using our best system. Then each SS candidate is labeled by human as acceptable or not. The evaluation is carried out using top-1 and top-10 results based on top-n inclusion rate. Top-n inclusion rate is defined as the percentage of the test sentences whose top-n outputs contain at least one acceptable SS. The Top-1 inclusion rate is 0.21 and the Top-10 inclusion rate is 0.73. The numbers are very encouraging for this difficult task. An analysis on the 27 FSs whose top-10 outputs contain no acceptable SS shows that the errors mainly come from three aspects: unidentified named entity, complicated character decomposition and repetition.

5 Model of Chinese Quatrain

We will describe our work on extension the model of Chinese couplets to Chinese classic poetry using quatrain that is composed with 4 sentences as case study. We believe that 8 sentence regulated verse can be generated with same approach.

Because the Chinese couplet can be considered a special kind of poetry with only two sentences, and since pairs of adjacent sentences in quatrains are quite similar to Chinese couplets, we are inspired to apply the same statistical MT model based approach. A necessary extension is made for generating the next sentence given a previous sentence. Specifically, when generating the second sentence, we regard the first sentence as the source language sentence in a MT task and regard the second sentence as the target language sentence. And similar approach is applied for generating the third sentence and the fourth sentence respectively. As in couplet generation, the SMT model we use here does not include word deletion, insertion and reordering, because quatrain sentences all have an identical number of characters. However, to meet the special requirements of poetic sentence generation, we made some necessary adaptations to the couplet generation model. First, to improve the coherence of non-adjacent sentences, we add a mutual information feature which takes all previously generated sentences into consideration and so helps to guarantee the cohesion between the sentence to be generated and all previously generated sentences. Second, to reflect the slightly different relationship between two adjacent sentences at different positions of a poem, when generating a sentence at a certain position, we combine the translation model trained on the data at the corresponding position with a background model trained on all poetic sentence pairs.

Different with the couplet generation process which the first sentence is given as input, the quatrain system receives no first sentence as input but a few key words that describe the topic,

from which the system generates the first sentence. As a preliminary experiment of this paper, to simplify the task, we limit the user-input key words for specifying topic to selections from a poetic phrase taxonomy "ShiXueHanYing" (诗学含英) which collected and classified phrases used in ancient Chinese poems. In the taxonomy, 41,218 phrases (34,290 unique), with length from 1 to 5 characters, are classified into 1,016 clusters. Each cluster is named after a concept keyword, such as "spring", or "mountaineering", where words associated to a concept is put together. The 1,016 keywords cover most of the frequent topics in ancient Chinese poems.

After the user gives a few keywords, the quatrain generator will generate the first sentence. Then given the first sentence, it generates the second sentence. Then given the third sentence, it generates the fourth sentence. Every step follows a similar SMT model to couplet generation. In the next section, we will describe the mechanism of generation of the first sentence. For SMT model and training data, as it is similar to the couplet system, readers can refer to Section 3.

6 Generation of the First Sentence

A template-based method is adopted to generate the first sentence with the input keywords. Based on the rhyme knowledge of Chinese classic poetry, the template of a 5-character poem sentence must be one of the following types: **|***, ***|**, *|****, ****|*, …, *|**|**, **|**|*. Here "**|***" represents the juxtaposition of a double-syllable phrase and a triple-syllable phrase. Other notations can be deduced similarly. The templates of 7-character sentences can be enumerated similarly. Assuming that all phrases in the first sentence of the poem are relevant to the key words given by the user, we can get a lot of first sentence candidates by filling all the pertinent phrases into the above templates.

Because we have constrained the user to select keywords from the cluster names in the taxonomy, it is straightforward to get the phrases related to the selected keywords after the user completes the selection. Then we randomly combine the phrases based on the templates listed above to form candidates for first sentences of the user-specified length. Let us explain the generation process with an example in Table 5. Suppose a user chooses three key words "春日" (spring day), "郊行" (outing) and "访友" (visiting friends), then specifies a 5-character Quatrain. Our system will first construct a phrase set from the ShiXueHanYing taxonomy related to the given key words, say {"明媚", "寻芳草", "水村", "旧话",…}, and put them at any possible position in the sentence to generate the lattice below:

Table 5: An example lattice

char 1	char 2	char 3	char 4	char 5
明媚		寻芳草		
晴光		鱼	蝶飞	
花变新红				绽
江山丽			迎门	
…	…	…	…	…

Then we search using the Forward-Viterbi-backward-A* algorithm and find the N best first sentence candidates such as "晴光寻芳草", "晴光鱼迎门", "江山丽蝶飞"…

7 Experimental Results

Finding an effective approach for automatic evaluation is a big challenge for poetry generation. It is difficult to use BLEU for the evaluation of poems because BLEU requires a reference set (i.e., sentence at $(n+1)^{th}$ line) for each sentence at n^{th} line. This requires a human-authored standard answer. However, given the same keywords, the poems generated by human authors can be too diverse to be able to enumerate. This kind of answer data base is not presently

available. Therefore, we adopted subjective evaluation of the generated poems. To better understand the performance of our method, we designed the following three experiments, each of which has detailed criteria for evaluating different parts of the generated poems.

7.1 Evaluation of the First Sentence

In this experiment, we prepare 40 groups of key words. Each group consists of three key words which are randomly selected from the taxonomy of "ShiXueHanYing". From the system generated sentences, two annotators were asked to select the best first sentence according to three criteria: the relatedness between this sentence and the input keywords, the fluency of this sentence and the appropriateness of this sentence to be the first sentence of a poem. For each criterion, the annotator will score the sentence with one of the three numbers (good=100, accept=50 and poor=0). We set the weights for the three criteria as 0.4, 0.4 and 0.2 respectively. For example, if a sentence gets 50, 100 and 100 for the three criteria respectively, this sentence will get a final score of 80. After the annotators label all best first sentences, we compute the average scores for 7-character and 5-character sentences separately, as shown below.

Table 6: Results of the first sentence evaluation

Form	5-character	7-character
Average score	69.5	78.5

As shown in Table 6, both of the scores are above 60, which means that the overall quality of the generated first sentences is acceptable. We also note that the result for the 7-character sentences is better than 5-character sentences. This is because a longer sentence is more likely to cover the meaning of the input key words.

7.2 Evaluation of the Next Sentence

We used the same keywords as in the previous experiment and the best first sentences that human selected. Then we generated 10 best second, third and fourth sentences in turn with the proposed method. After every generation, we manually selected one best sentence and treated it as input to generate the next sentence. Finally, we obtained 40 groups of generated second, third and fourth sentences. Then annotators judge their quality based on two criteria: Fluency and Relatedness, with equal weight of 0.5. For each criterion, the annotator will score the sentence with one of the three numbers (good=100, accept=50 and poor=0). After the judgment, we calculate the score of each sentence, by which sentences are classified into the following three classes: Sentences with score>=50 are Acceptable, sentences with score=100 are Perfect and sentences with score=0 are Poor. Then we calculate the percentage of test cases whose the top N outputs contain at least one perfect sentence or acceptable sentence. The results for 5-character poems and 7-character poems are listed in Table 7.

Table 7: Result for (a) 5- and (b) 7-character sentences

(a)

	Perfect			Acceptable		
	Top 1	Top 5	Top10	Top 1	Top 5	Top10
1^{st}->2^{nd}	15%	37%	50%	65%	87%	90%
2^{nd}->3^{rd}	20%	43%	53%	65%	93%	97%
3^{rd}->4^{th}	15%	37%	47%	35%	80%	87%

(b)

	Perfect			Acceptable		
	Top 1	Top 5	Top10	Top 1	Top 5	Top10
1^{st}->2^{nd}	0	40%	53%	65%	80%	87%
2^{nd}->3^{rd}	20%	44%	50%	50%	63%	75%
3^{rd}->4^{th}	10%	39%	44%	45%	67%	78%

From Table 7, we can see that the system is able to get at least an acceptable sentence as a top 1 result for about 50% of cases. When we check the top-5 and top-10 sentences, the acceptable rate rises to 80%. However, the overall rate of perfect sentences is much smaller than that of the acceptable level, indicating that our system can generate poetic sentences that roughly meet the rules but is still hard to generate excellent ones.

7.3 Evaluation of the Whole Poem

To evaluate the quality of the whole generated poem, we randomly selected 20 groups of key words to generate 20 poems (ten 5-character and ten 7-character) with our system. When generating poems with our system, we use two different settings: one is interactive mode which means at each step of the generation we manually select one best generated sentence and regard it as input to generate the next sentence; the other is total automation which means at each step the system generates only one best output. After the poems are generated, annotators score them with the following criteria.

Table 8: Criteria for the whole poem evaluation

Criteria	Weight	Poor	Acceptable	Good
Fluency	5/15	0	50	100
Rhyme	5/15	0	50	100
Relatedness	3/15	0	50	100
Structure	3/15	0	50	100
Artistic conception	1/15	0	50	100

After scoring, we calculated the score of each poem and get the average score of the 20 poems for each system. The Automatic Mode achieves 68.43% average score and the Interactive Mode achieves 77.83% average score. This means with Interactive Mode the quality can be largely improved. To get intuitive understanding, the following is a poem generated with the keywords "踏青 赏花 游春" (outing, enjoy the blossom, spring sightseeing) by our system, with the setting of total automation. 回舟一水香醉月/落日千山雪吟风/踏青寻花问柳春/人不在酒云梦中.

8 Related Work

In the area of computer-assisted Chinese poetry generation research, (Lo et al., 1999) has developed a tool which provides the rhyme templates for various styles of ancient Chinese poetry and a dictionary for tone specification of any Chinese character. Users can write their own poems with the help of this tool. The rhyme templates and the dictionary were human-authored and hand-compiled.

A prior attempt at automatic Chinese poem generation, the "Daoxiang" poem generator (http://www.poeming.com/web/index.htm) system, directly fills classic rhyme templates (chosen from a large inventory) with words generally used in the ancient poems. The poems thus generated, although formally fitting the necessary rhythmic and metrical constraints, generally cannot support a satisfactory reader experience because: 1) usually the generated sentences are not fluent and have unclear meanings; 2) the generated sentences seem independent of each other, which makes it difficult for readers to derive a holistic meaning from the poem.

For other languages, approaches for creating poetry with computers began in 1959 when Theo Lutz created the first example of "Computer Poetry" in Germany. Masterman (1971) and Tosa et al. (2008) describe two haiku producers. Other systems include RACTER and PROSE (Hartman, 1996). Approaches to poetry generation can roughly be classified into template-based, evolutionary, and case-based reasoning. Typically, for the template-based approach, the

generation process randomly chooses words from a hand-crafted lexicon and then fills in the slots provided by a template-based grammar.

To the best of our knowledge, there is no existing research work published on statistical methods for generation of Chinese poems and couplets. Our work can be regarded as the first attempt using statistical approach in this field.

9 Conclusions and Future Work

We propose a novel statistical approach to automatically generate Chinese couplets (http://duilian.msra.cn) and Chinese poetry. This design well incorporates the statistical approach and linguistic constraints of Chinese couplets and poetry. A comprehensive evaluation has been conducted for couplets generation and for quatrain generation and promising results were achieved.

In the future, we will explore an automatic evaluation approach of each individual line and the whole poetry of generated quatrain. Besides, to enhance thematic structure of a generated poetry, it is important to study to a better method to incorporate the contents of all of the generated sentences for the generation of the current line.

References

Fan, C., L. Jiang, M. Zhou and S.-L. Wang. 2007. Mining Collective Pair Data from the Web. In *Proc. of the International Conference on Machine Learning and Cybernetics 2007*, pages 3997-4002.

Hartman, Charles O. 1996. *Virtual Muse: Experiments in Computer Poetry*. Wesleyan University Press.

He, J., M. Zhou and L. Jiang. To appear. Generating Chinese Poems using a Statistical MT Approach. *Journal of Chinese Information Processing* (in Chinese).

Jiang, L. and M. Zhou. 2008. Generating Chinese Couplets using a Statistical MT Approach. In *Proceedings of the 22nd International Conference on Computational Linguistics*, Manchester, England.

Koehn, P., F. J. Och and D. Marcu. 2003. Statistical phrase-based translation, In *Proceedings of HLT-NAACL 2003*, pages 48-54.

Liu, Wenwei. 1735. *ShiXueHanYing* (诗学含英).

Lo, Feng-Ju, Yuan-Ping Lee and Wei-Cheng Tsao. 1999. The Format Auto-Checking and Database Indexing Teaching System of Chinese Poetry and Lyrics. *Journal of Chinese Information Processing*, Vol. 13, No. 1, pp. 35-42.

Masterman, M. 1971. Computerized Haiku. *Cybernetics*, pp. 175-183.

Och, F. J. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the 41st Annual Meeting of the Association for Computational Linguistics*.

Och, F. J. and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*.

Och, F. J. and H. Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30:417-449.

Papineni, K., S. Roukos, T. Ward and W.-J. Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proc. of the 40th Annual Meeting of the Association for Computational Linguistics*.

Stolcke, Andreas. 2002. SRILM -- An Extensible Language Modeling Toolkit. In *Proc. of Intl. Conf. on Spoken Language Processing*, vol. 2, pp. 901-904.

Tosa, Naoko, Hideto Obara and Michihiko Minoh. 2008. Hitch Haiku: An Interactive Supporting System for Composing Haiku Poem. *ICEC 2008*, pp. 209-216.