

# Generating Chinese Couplets using a Statistical MT Approach

**Long Jiang**

Microsoft Research Asia  
Sigma Center, No. 49, Zhichun Road  
Haidian District, Beijing 100190, PRC  
longj@microsoft.com

**Ming Zhou**

Microsoft Research Asia  
Sigma Center, No. 49, Zhichun Road  
Haidian District, Beijing 100190, PRC  
mingzhou@microsoft.com

## Abstract

Part of the unique cultural heritage of China is the game of Chinese couplets (du li ǎn). One person challenges the other person with a sentence (first sentence). The other person then replies with a sentence (second sentence) equal in length and word segmentation, in a way that corresponding words in the two sentences match each other by obeying certain constraints on semantic, syntactic, and lexical relatedness. This task is viewed as a difficult problem in AI and has not been explored in the research community.

In this paper, we regard this task as a kind of machine translation process. We present a phrase-based SMT approach to generate the second sentence. First, the system takes as input the first sentence, and generates as output an N-best list of proposed second sentences, using a phrase-based SMT decoder. Then, a set of filters is used to remove candidates violating linguistic constraints. 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.

## 1 Introduction

Chinese antithetical couplets, called “du li ǎn”, form a special type of poetry composed of two

sentences. They use condensed language, but have deep and sometimes ambivalent meanings. The two sentences making up the couplet are called the “first sentence” (FS) and the “second sentence” (SS) respectively.

Chinese couplets are considered an important cultural heritage. A couplet is often written in calligraphy on vertical red banners, and typically placed on either side of a door or in a large hall during special occasions such as wedding ceremonies and the Chinese New Year. People also use couplets to celebrate birthdays, mark the openings of a business, and commemorate historical events. Chinese couplets have also been used effectively in teaching Chinese in China.

An example of a Chinese couplet is “海阔凭鱼跃; 天高任鸟飞”, where the FS is “海阔凭鱼跃” and the SS is “天高任鸟飞”. It says that the sea is wide enough so that fish can jump at their pleasure, and the sky is high enough so that bird can fly unrestrictedly. The correspondence between individual words of the FS and SS is shown here:

海	阔	凭	鱼	跃
sea	wide	allow	fish	jump
天	高	任	鸟	飞
sky	high	permit	bird	fly

Figure 1. An Example of a Chinese Couplet.

Generating the SS of a Chinese couplet given the FS can be viewed as a big challenge in AI. As far as we know, there is no previous work to tackle this problem.

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 se-

© 2008. Licensed under the *Creative Commons Attribution-Noncommercial-Share Alike 3.0 Unported* license (<http://creativecommons.org/licenses/by-nc-sa/3.0/>). Some rights reserved.

lected 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 SMT model is applied to generate an 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.

We implemented a web service based on our approach (anonymous URL). A user can input a FS, and our software outputs its top 10 best-scoring SS candidates. Tens of thousands of people use our service every day.

The rest of the paper is organized as follows. In Section 2, we explain the motivation of our work. Then Sections 3 and 4 detail our multi-phase SMT approach for the SS generation. The experimental results and evaluation are reported in Section 5 and related work on computer poetry is summarized in Section 6. In Section 7, we conclude our study and point out the future work.

## 2 Motivation

Chinese couplets vary widely in length. A short couplet could consist of two sentences each containing only one or two characters while a longer couplet may reach several hundred characters. However, the length of sentences in most Chinese couplets is between 5 and 10 characters. There are also diverse forms of writing couplets. For instance, in one form, the FS and SS are similar in meaning, while in another, they have to oppose in meaning.

However, no matter which form a couplet follows, it generally must conform to the following constraints:

**Constraint 1:** The two sentences of a couplet agree in length and word segmentation. For example, if a FS contains 7 characters and the first two characters form a word, then the qualified SS should also contain 7 characters with the first two forming a word.

**Constraint 2:** Tones are generally “coinciding and harmonious”: In Chinese, every character is pronounced either “Ping” (平) or “Ze” (仄). In a Chinese couplet, the character at the end of the FS should be “Ze” (pronounced in a sharp

downward tone); the character at the end of the SS should be “Ping” (pronounced in a level tone).

**Constraint 3:** Corresponding words in the two sentences should agree in their part of speech and characteristics. For instance, a noun in the SS should correspond to a noun at the same position in the FS. A named entity should correspond to a named entity.

**Constraint 4:** The contents of the two sentences should be related, but not duplicated.

**Constraint 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. And if there is a character decomposition in the FS, i.e., the FS contains a character and its “component characters”, the SS should contain a character decomposition at the corresponding positions.

Character decomposition is an interesting language phenomenon in Chinese: some Chinese characters can be decomposed into other characters. For example, “好” (good) can be decomposed into “女” (daughter) and “子” (son). As illustrated in Figure 2, the left part of “好” is “女” and the right part is “子”. “女” and “子” are called the “component characters” of “好”.

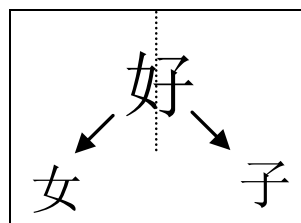


Figure 2. Character Decomposition.

Compared to western couplets, which also consist of two sentences that usually rhyme and have the same number of syllables, Chinese couplets have much stronger constraints. Because in Chinese each character has one and only one syllable, the same number of syllables means the same number of characters. Moreover, the constraints of the FS and SS on consistency of part of speech sequence and writing style make Chinese couplets have more regular form.

Given the FS, writing a good SS to match it is a difficult task because the SS must conform to constraints on syntax, rhyme and semantics, as described above. It also requires the writer to innovatively use extensive knowledge in different disciplines. Some of the difficulties can be seen from the following example:

有	女	有	子	方	称	好
have	daughter	have	son	so	call	good
缺	鱼	缺	羊	敢	叫	鲜
lack	fish	lack	mutton	dare	call	delicious

Figure 3. An Example of a Complicated Couplet.

Figure 3 shows a complicated couplet of “有女有子方称好; 缺鱼缺羊敢叫鲜” (Once one has a daughter and son, one’s life is complete; who would dare call a meal without fish and mutton delicious? In China, there is an old saying that courses made of fish and mutton are most delicious). The FS contains a repeated character “有” (have), and a character decomposition: “好” (good) and its “component characters” “女” (daughter) and “子” (son). So it requires that the qualified FS should contain identical character repetition and character decomposition. A perfect SS worked out after multiple attempts by many people for this FS is “缺鱼缺羊敢叫鲜”, which equally contains a repeated character “缺” (lack), and a character decomposition: “鲜” (fresh) and its “component characters” “鱼” (fish) and “羊” (mutton) at the corresponding positions. And the meanings of the two sentences are also parallel: they tell us what is important in life and what is important in cuisine, respectively.

### 3 Couplet Generation Model

In this paper, a multi-phase 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 recent reranking approaches of SMT (Och and Ney, 2004). In the 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 candidates of low quality. 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  $F = \{f_1, f_2, \dots, f_n\}$ , our objective is to seek a SS denoted as

$S = \{s_1, s_2, \dots, s_n\}$ , where  $f_i$  and  $s_i$  are Chinese characters, so that  $p(S/F)$  is maximized. Following Och and Ney (2002), we depart from the traditional noisy-channel approach and use a more general log-linear model. Then the  $S^*$  that maximizes  $p(S/F)$  can be expressed as follows:

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

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 like traditional Chinese and most of words contain only one character. If we try to incorporate Chinese word segmentation, it may bring in unexpected errors. However, we will still report the comparison to the word-based method in Subsection 5.3.

Among features commonly used in phrase-based SMT, five features, listed in Table 1, were selected for our model. To apply phrase-based features,  $S$  and  $F$  are segmented into phrases  $\bar{s}_1 \dots \bar{s}_l$  and  $\bar{f}_1 \dots \bar{f}_l$ , respectively. We assume a uniform distribution over all possible segmentations.

$h_1(S, F) = \prod_{i=1}^l p(\bar{f}_i   \bar{s}_i)$	Phrase translation model
$h_2(S, F) = \prod_{i=1}^l p(\bar{s}_i   \bar{f}_i)$	Inverted phrase translation model
$h_3(S, F) = \prod_{i=1}^l p_w(\bar{f}_i   \bar{s}_i)$	Lexical weight
$h_4(S, F) = \prod_{i=1}^l p_w(\bar{s}_i   \bar{f}_i)$	Inverted lexical weight
$h_5(S, F) = p(S)$	Language model

Table 1. Features in our SMT Model.

#### Phrase translation model (PTM)

In a phrase-based SMT model, phrases can be any substring that may not necessarily be linguistically motivated. 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 corresponding words in the FS and SS, respectively. As a result, the  $i^{\text{th}}$  character/phrase in  $F$  is exactly “translated” into the  $i^{\text{th}}$  character/phrase in  $S$ . Based on this rule, the phrase translation probability  $p(\bar{f}_i | \bar{s}_i)$  can be estimated by relative frequency on a training corpus:

$$p(\bar{f}_i | \bar{s}_i) = \frac{\text{count}(\bar{f}_i, \bar{s}_i)}{\sum_{r=1}^m \text{count}(\bar{f}_r, \bar{s}_i)} \quad (2)$$

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

The inverted phrase translation model  $p(\bar{s}_i | \bar{f}_i)$  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  $p_w(\bar{f}_i | \bar{s}_i)$ , which indicates how well its words translate to each other:

$$p_w(\bar{f}_i | \bar{s}_i) = \prod_{j=1}^{N_i} p(f_j | s_j) \quad (3)$$

where  $N_i$  is the number of characters in  $\bar{f}_i$  or  $\bar{s}_i$ ,  $f_j$  and  $s_j$  are characters in  $\bar{f}_i$  and  $\bar{s}_i$  respectively, and  $p(f_j | s_j)$  is the character translation probability of  $s_j$  into  $f_j$ . Like in phrase translation probability estimation,  $p(f_j | s_j)$  can be computed by relative frequency:

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

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

Like for the phrase translation model, we also use an inverted lexical weight  $p_w(\bar{s}_i | \bar{f}_i)$  in addition to the conventional lexical weight  $p_w(\bar{f}_i | \bar{s}_i)$  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 Model Training

A Chinese couplet corpus is necessary for estimating the phrase and character translation probabilities. Currently, there is, however, no large-sized Chinese couplet collection available. Based on our observation, there are many pages on the web containing classic Chinese couplets collectively. So we used the method proposed by (Fan et al., 2007) to recursively mine those couplets with the help of some seed couplets. The method can automatically learn patterns in a page which contains collectively Chinese couplets and then apply the learned pattern to extract more Chinese couplets. There are also some online forums where Chinese couplet fans meet. When some people post FSs on the forums, many other people submit their SSs in response. Such data seems useful for our model training. So we crawled all posted FSs with all their replied SSs. Then from the crawled data, FSs having over 20 unique SSs are selected as development or testing set (see Subsection 5.1), and others are used for model training. Finally, with web mining approach, we collected 670,000 couplets.

To enhance the couplet database crawled from the web, 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-sentence Tang poetry, the third and fourth sentences and the fifth and sixth sentences form pairs basically satisfying the constraints of Chinese couplets. Therefore, these sentence pairs can be used as couplets in our training algorithm. In that way we get additional 300,000 sentence pairs 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.

For the language people use in Chinese couplets is same as that in Chinese poetry, for the purpose of smoothing the language model we add about 1,600,000 sentences from ancient Chinese poetry to train language model, which are not necessarily couplets.

To estimate the weights  $\lambda_i$  in formula (1), we use 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 5.1.

## 4 Couplet Generation

In this section, we will detail each step of the generation of the second sentence.

### 4.1 Decoding for N-best Candidates

First, we use a phrase-based decoder similar to the one by (Koehn et al., 2003) to generate an N-best list of SS candidates. Because there is no word reordering operation in the SS generation, our decoder is a monotonic decoder. In addition, the input FS is often shorter than ordinary MT input sentence, so our decoder is more efficient.

### 4.2 Linguistic Filters

A set of filters is used to remove candidates that violate linguistic constraints that well-formed Chinese couplets should obey.

#### Repetition filter

This filter removes candidates based on various rules related to word or character repetition. One such rule requires that if there are characters that are identical in the FS, then the corresponding characters in the SS should be identical too. For example, in a FS “有女有子方称好” (have daughter have son so call good), the word “有” is repeating. The legal SS should also contain corresponding repeating words. For instance, a qualified second sentence “缺鱼缺羊敢叫鲜” (lack fish lack mutton dare call delicious) would be legal because “缺” corresponds to “有” and is repeating in the same way. Conversely, if there are no identical words in the FS, then the SS should have no identical words.

#### Pronunciation repetition filter

This filter works similarly to the repetition filter above except it checks the pronunciation of characters not the character surfaces. The pronunciation of a character can be looked up from a Chinese character pronunciation dictionary. For simplicity, we only use the first pronunciation in the dictionary for polyphones.

#### Character decomposition filter

We compiled a Chinese character decomposition table from which one can look up what characters a Chinese character can be decomposed into. The decomposition information can be derived from the strokes of each character in a dictionary and then verified by human. Based on this table, we can easily filter out those SS candidates which contain different character decompositions at the corresponding positions from the FS.

#### Phonetic harmony filter

We filter out the SSs with improper tones at the end character position according to the Chinese character pronunciation dictionary.

### 4.3 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 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),  $\bar{x}$  is the feature vector of a SS candidate, and  $\bar{w}$  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.

$$f_{\bar{w}}(\bar{x}) = \langle \bar{w}, \bar{x} \rangle \quad (5)$$

Besides the five features used in the phrase-based SMT model, additional features for reranking are as follows:

#### 1. Mutual information (MI) score:

This feature is designed to measure the semantic consistency of words in a SS candidate. For example, the two candidates “天高任鸟飞” (sky high permit bird fly) and “天高任狗叫” (sky high permit dog bark) have similar PTM, LW and LM scores. However, human beings recognize the former as a better phrase, because “天高” (sky high) and “狗叫” (dog bark) in the latter sentence do not make any sense together. MI can capture the associations between words, whether they are adjacent or not.

Specifically, given a SS candidate  $S = \{s_1, s_2, \dots, s_n\}$ , we use the following formula to compute the MI score:

$$MI(S) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n I(s_i, s_j) = \sum_{i=1}^{n-1} \sum_{j=i+1}^n \log \frac{p(s_i, s_j)}{p(s_i)p(s_j)} \quad (6)$$

The parameters  $p(s_i, s_j)$ ,  $p(s_i)$  and  $p(s_j)$  are estimated using Maximum Likelihood Estimation on the same training data as for training PTM.

#### 2. MI-based structural similarity (MISS) score:

In a Chinese couplet, if two words in the FS are strongly associated, their corresponding words in the SS should also be strongly associated, and vice versa. For example, in the couplet “海阔凭鱼跃; 天高任鸟飞” (sea wide allow fish jump; sky high permit bird fly), the word pairs “海”

(sea) and “阔” (wide), “海” (sea) and “鱼” (fish), “鱼” (fish) and “跃” (jump) in the FS are all strongly associated. Similarly, the corresponding word pairs “天” (sky) and “高” (high), “天” (sky) and “鸟” (bird), “鸟” (bird) and “飞” (fly) in the SS are all strongly associated. To measure this kind of structural similarity, we develop a measure function called MI-based structural similarity score. Specifically, given the FS  $F = \{f_1, f_2, \dots, f_n\}$ , we first build its vector  $V_f = \{v_{12}, v_{13}, \dots, v_{1n}, v_{23}, \dots, v_{n-1n}\}$ , where  $v_{ij}$  is the mutual information of  $f_i$  and  $f_j$  (i.e.,  $I(f_i, f_j)$  in formula (6)). Then we build a vector  $V_s$  for each SS candidate in the same way. We use a cosine function to compute the similarity between the two vectors as the MISS score:

$$MISS(F, S) = \cos(V_f, V_s) = \frac{V_f \cdot V_s}{|V_f| \times |V_s|} \quad (7)$$

To estimate the parameter vector in the Ranking SVM model, we used an existing training tool, SVM Light<sup>2</sup>, and a labeled training corpus. We selected 200 FSs with a length of 7 or 8 characters. For each of them, 50 SS candidates are generated using the N-best SMT decoder. Two operators are asked to label each SS candidate as positive if the candidate is acceptable and as negative if not. After removing 10 FSs and their SS candidates as they had no corresponding positive SS, we got 190 FSs with 9,500 labeled SS candidates (negative: 6,728; positive: 2,772) to train the Ranking SVM model.

## 5 Experimental Results

### 5.1 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 can be 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 stands for better translation quality.

$$BLEU = BP \cdot \exp\left(\sum_{n=1}^N w_n \log p_n\right) \quad (8)$$

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 less 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 removing some noisy references by human, 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.

### 5.2 BLEU vs. Human Evaluation

To justify whether BLEU is suitable for evaluating generated SSs, we compare BLEU with human evaluation. Figure 4 shows a linear regression of the human evaluation scores as a function of the BLEU score for our 6 systems which generate SSs given FSs. Among the 6 systems, three are implemented using a word-based SMT model with 100K, 400K, and 970K couplets for training, respectively, while the other three are implemented using a phrase-based SMT model with 100K, 400K, and 970K couplets for training, respectively. The word-based SMT model contains only two features: word translation model and language model. The word translation model is trained on the corpus segmented by a Chinese word breaker implemented by (Gao et al., 2003). We selected 100 FSs from the testing data set; for each of them, the best SS candidate was generated using each system. Then we computed the BLEU score and the human score of each system.

The human score is the average score of all SS candidates. Each candidate is scored 1 if it is acceptable, and 0 if not. The correlation of 0.92 indicates that BLEU tracks human judgment well.

<sup>2</sup> <http://svmlight.joachims.org/>

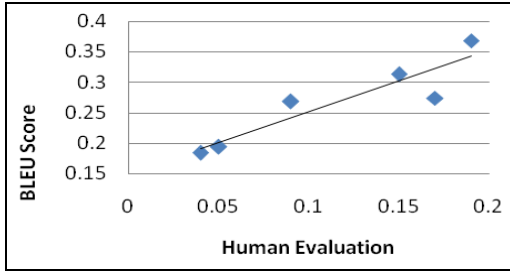


Figure 4: BLEU Predicts Human Judgments.

### 5.3 Translation Unit Setting

We conducted some experiments to compare the system performances with different translation unit settings: character-based, word-based and phrase-based. In each setting, we only use translation probability and language model as features. And after SMT decoder, we use the same filtering but no reranking. We use all 451 testing data and the results are listed below:

Translation Unit setting	BLEU
character-based	0.236
word-based	0.261
phrase-based	0.276

Table 2. Different Translation Unit Setting.

As shown in Table 2, the word-based translation model achieves 0.025 higher of BLEU score than the character-based model. And the phrase-based model gets the highest score. The improvement shows that phrase-based model works better than word-based and character-based model in our task of SS generation.

### 5.4 Feature Evaluation

We also 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.

	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

Table 3. Feature Evaluation.

As shown in Table 3, 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. Furthermore, with the Ranking SVM model, the score is improved by 0.13 percent, from 0.348 to 0.361. This means our re-ranking model is helpful.

### 5.5 Overall Performance Evaluation

In addition to the BLEU evaluation, we also carried out human evaluation. We select 100 FSs from the log data of our couplet web service mentioned in Section 1. 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 results are listed below:

	Top-1	Top-10
<i>Top-n inclusion rate</i>	0.21	0.73

Table 4. Overall Performance Evaluation.

As shown in Table 4, our system can get a 0.21 of top-1 inclusion rate and 0.73 of top-10 inclusion rate. The numbers seem a little low, but remember that generating a SS given a FS is a quite difficult job, and even humans cannot do it well in limit time, for example, 5 minute per FS. However, what is more important is that our system can provide users diversified SSs and many unacceptable SSs generated by our system can be easily refined by users to become acceptable.

We also made careful analysis on the 27 FSs whose top-10 outputs contain no acceptable SS. As shown in Table 5, the errors mainly come from three aspects: unidentified named entity, complicated character decomposition and repetition. An example of complicated repetition is “近世 进士 尽是 近视” (modern /scholar/all/myopic, modern scholars are all myopic). In this sentence, the pronunciations of the four words are identical (j ñsh ñ), a qualified SS must be meaningful and posses same repetitions, which poses a big challenge to the system.

Mistake types	# of FS
Mistakes with named entities	6
Complicated character decomposition	5
Complicated repetition	4
Mistakes of miscellaneous types	12

Table 5. Error Analysis.

## 6 Related Work

To the best of our knowledge, no research has been published on generating the SS given the FS of a Chinese couplet. However, because our task can be viewed as generating the second line of a special type of poetry given the first line, we consider automatic poetry generation to be the most closely related existing research area.

As to computer-assisted Chinese poetry generation, Luo has developed a tool<sup>3</sup> which provides the rhyme templates of forms of classical Chinese poetry and a dictionary in which one can look up the tone of a Chinese character. Both the rhyme templates and the dictionary were compiled by human efforts.

For other languages, approaches to creating poetry with computers began in 1959 when Theo Lutz created the first examples of “Computer Poetry” in Germany (Hartman, 1996). Masterman finished a haiku producer (Manurung et al., 2000). 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 gaps provided by a template-based grammar. In computer poetry systems, the starting point is a given message, or communicative goal, and the aim is to produce a string of text that conveys that message according to the linguistic resources available.

There is a big difference between our task and poetry generation. When generating the SS of a Chinese couplet, the FS is given. The task of generating the SS to match the FS is more well-defined than generating all sentences of a poem. Furthermore, the constraints on Chinese couplets mentioned above will enable us to do a more objective evaluation of the generated SSs.

## 7 Conclusions and Future Work

This paper presents a novel approach to solve the problem of generating Chinese couplets. An SMT approach is proposed to generate the SSs for a FS of a Chinese couplet. The system is comprised of a phrase-based SMT model for the generation of an N-best list of SS candidates, a set of linguistic filters to remove unqualified candidates to meet the special constraints of Chinese couplets, and a discriminative reranking

model incorporating multi-dimensional features to get better results. The experimental results show that this approach is very promising.

As a future work, it would be interesting to investigate how this approach can be used in poetry generation.

## References

- P. F. Brown, S. A. Della Pietra, V. J. Della Pietra and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19:2, 263-311.
- David Chiang. 2005. A Hierarchical Phrase-Based Model for Statistical Machine Translation. In *Proc. of the 43<sup>rd</sup> Meeting of the Association for Computational Linguistics*, pages 263-270.
- B. Dáz-Agudo, P. Gervás and P. González-Calero. 2002. Poetry generation in COLIBRI. In *Proc. of the 6<sup>th</sup> European Conference on Case Based Reasoning*, Aberdeen, Scotland.
- C. Fan, L. Jiang, M. Zhou, 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.
- Jianfeng Gao, Mu Li and Changning Huang. 2003. Improved source-channel models for Chinese word segmentation. In *Proc. of the 41<sup>st</sup> Meeting of the Association for Computational Linguistics*.
- Charles O. Hartman. 1996. *Virtual Muse: Experiments in Computer Poetry*. Wesleyan University Press.
- P. Koehn, F. J. Och and D. Marcu. 2003. Statistical phrase-based translation, In *HLT-NAACL 2003*, pages 48-54.
- H. Manurung, G. Ritchie and H. Thompson. 2001. Towards a computational model of poetry generation. In *Proc. of the AISB-00 Symposium on Creative and Cultural Aspects of AI*, 2001.
- F. J. Och. 2003. Minimum error rate training in statistical machine translation. In *Proc. of the 41<sup>st</sup> Meeting of the Association for Computational Linguistics*.
- F. J. Och and H. Ney. 2002. Discriminative training and maximum entropy models for statistical machine translation. In *Proc. of the 40<sup>th</sup> Meeting of the Association for Computational Linguistics*.
- F. J. Och and H. Ney. 2004. The Alignment Template Approach to Statistical Machine Translation. *Computational Linguistics*, 30:417-449.
- K. Papineni, S. Roukos, T. Ward and W.-J. Zhu. 2002. BLEU: a Method for automatic evaluation of machine translation. In *Proc. of the 40<sup>th</sup> Meeting of the Association for Computational Linguistics*.

<sup>3</sup> <http://cls.hs.yzu.edu.tw>