

Chinese Couplet Generation with Neural Network Structures

Rui Yan^{1,2}, Cheng-Te Li³, Xiaohua Hu⁴, and Ming Zhang⁵

¹Institute of Computer Science and Technology, Peking University, Beijing 100871, China

²Natural Language Processing Department, Baidu Inc., Beijing 100193, China

³Academia Sinica, Taipei 11529, Taiwan

⁴College of Computing and Informatics, Drexel University, Philadelphia, PA 19104, USA

⁵Department of Computer Science, Peking University, Beijing 100871, China

yanrui02@baidu.com, ctli@citi.sinica.edu.tw

xh29@drexel.edu, mzhang_cs@pku.edu.cn

Abstract

Part of the unique cultural heritage of China is the Chinese couplet. Given a sentence (namely an *antecedent clause*), people reply with another sentence (namely a *subsequent clause*) equal in length. Moreover, a special phenomenon is that corresponding characters from the same position in the two clauses match each other by following certain constraints on semantic and/or syntactic relatedness. Automatic couplet generation by computer is viewed as a difficult problem and has not been fully explored. In this paper, we formulate the task as a natural language generation problem using neural network structures. Given the issued antecedent clause, the system generates the subsequent clause via sequential language modeling. To satisfy special characteristics of couplets, we incorporate the attention mechanism and polishing schema into the encoding-decoding process. The couplet is generated incrementally and iteratively. A comprehensive evaluation, using perplexity and BLEU measurements as well as human judgments, has demonstrated the effectiveness of our proposed approach.

1 Introduction

Chinese antithetical couplets, (namely “对联”), form a special type of poetry composed of two clauses (i.e., sentences). The popularity of the game of Chinese couplet challenge manifests itself in many aspects of people’s life, e.g., as a means of expressing personal emotion, political views, or communicating messages at festive occasions. Hence, Chinese couplets are considered an important cultural heritage. A couplet is often written

in calligraphy on red banners 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. We illustrate a real couplet for Chinese New Year celebration in Figure 1, and translate the couplet into English character-by-character.

Usually in the couplet generation game, one person challenges the other person with a sentence (namely an *antecedent clause*). The other person then replies with another sentence (namely a *subsequent clause*) equal in length and term segmentation, in a way that corresponding characters from the same position in the two clauses match each other by obeying certain constraints on semantic and/or syntactic relatedness. We also illustrate the special phenomenon of Chinese couplet in Figure 1: “one” is paired with “two”, “term” is associated with “character”, “hundred” is mapped into “t-housand”, and “happiness” is coupled with “treasures”. As opposed to free languages, couplets have unique poetic elegance, e.g., aestheticism and conciseness etc. Filling in the couplet is considered as a challenging task with a set of structural and semantic requirements. Only few best scholars are able to master the skill to manipulate and to organize terms.

The Chinese couplet generation given the antecedent clause can be viewed as a big challenge in the joint area of Artificial Intelligence and Natural Language Processing. With the fast development of computing techniques, we realize that computers might play an important role in helping people to create couplets: 1) it is rather convenient for computers to sort out appropriate term combinations from a large corpus, and 2) computer programs can take great advantages to recognize, to learn, and even to remember patterns or rules given the corpus. Although computers are no sub-



Figure 1: An example of a Chinese couplet for Chinese New Year. We mark the character-wise translation under each Chinese character of the couplet so as to illustrate that each character from the same position of the two clauses has the constraint of certain relatedness. Overall, the couplet can be translated as: the term of “peaceful and lucky” (i.e., 和顺) indicates countless happiness; the two characters “safe and sound” (a.k.a., 平 and 安) worth innumerable treasures.

stitute for human creativity, they can process very large text repositories of couplets. Furthermore, it is relatively straightforward for the machine to check whether a generated couplet conforms to constraint requirements. The above observations motivate automatic couplet generation using computational intelligence. Beyond the long-term goal of building an autonomous intelligent system capable of creating meaningful couplets eventually, there are potential short-term applications for augmented human expertise/experience to create couplets for entertainment or educational purposes.

To design the automatic couplet generator, we first need to empirically study the generation criteria. We discuss some of the general generation standards here. For example, the couplet generally have rigid formats with the same length for both clauses. Such a syntactic constraint is strict: both clauses have exactly the same length while the length is measured in Chinese characters. Each character from the same position of the two clauses have certain constraints. This constraint is less strict. Since Chinese language is flexible sometimes, synonyms and antonyms both indicate semantic relatedness. Also, semantic coherence is a critical feature in couplets. A well-written couplet is supposed to be semantically coherent among both clauses.

In this paper we are concerned with automatic couplet generation. We propose a neural couplet machine (NCM) based on neural network structures. Given a large collection of texts, we learn representations of individual characters, and their

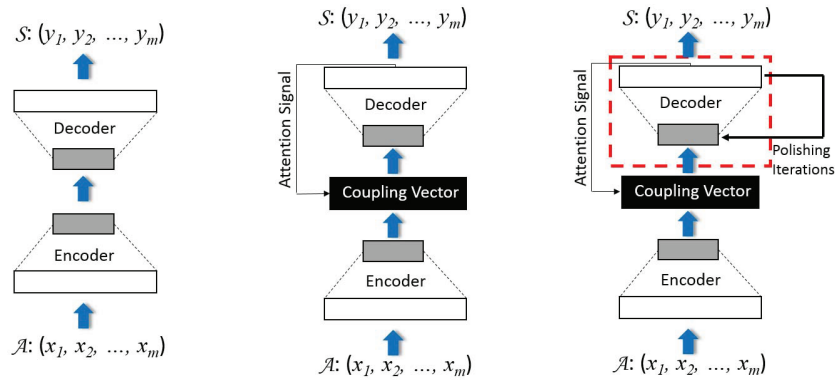
combinations within clauses as well as how they mutually reinforce and constrain each other. Given any specified antecedent clause, the system could generate a subsequent clause via sequential language modeling using encoding and decoding. To satisfy special characteristics of couplets, we incorporate the attention mechanism and polishing schema into the generation process. The couplet is generated incrementally and iteratively to refine wordings. Unlike the single-pass generation process, the hidden representations of the draft subsequent clause will be fed into the neural network structure to *polish* the next version of clause in our proposed system. In contrast to previous approaches, our generator makes utilizations of neighboring characters within the clause through an iterative polishing schema, which is novel.

To sum up, our contributions are as follows. For the first time, we propose a series of neural network-based couplet generation models. We formulate a new system framework to take in the antecedent clauses and to output the subsequent clauses in the couplet pairs. We tackle the special characteristics of couplets, such as corresponding characters paired in the two clauses, by incorporating the attention mechanism into the generation process. For the 1st time, we propose a novel polishing schema to iteratively refine the generated couplet using local pattern of neighboring characters. The draft subsequent clause from the last iteration will be used as additional information to generate a revised version of the subsequent clause.

The rest of the paper is organized as follows. In Section 2, we briefly summarize related work of couplet generation. Then Sections 3 and 4 show the overview of our approach paradigm and then detail the neural models. The experimental results and evaluation are reported in Section 5 and we draw conclusions Section 6.

2 Related Work

There are very few studies focused on Chinese couplet generation, based on templates (Zhang and Sun, 2009) or statistic translations (Jiang and Zhou, 2008). The Chinese couplet generation task can be viewed as a reduced form of 2-sentence poem generation (Jiang and Zhou, 2008). Given the first line of the poem, the generator ought to generate the second line accordingly, which is a similar process as couplet generation. We consider automatic Chinese poetry generation to be a closely re-



(a). Sequential couplet generation. (b). Couple generation with attention. (c). Couple generation with polishing schema.

Figure 2: Three neural models for couplet generation. More details will be introduced in Section 4.

lated research area. Note that there are still some differences between couplet generation and poetry generation. The task of generating the subsequent clause to match the given antecedent clause is more well-defined than generating all sentences of a poem. Moreover, not all of the sentences in the poems need to follow couplet constraints.

There are some formal researches into the area of computer-assisted poetry generation. Scientists from different countries have studied the automatic poem composition in their own languages through different ways: 1) Genetic Algorithms. Manurung *et al.* (2004; 2011) propose to create poetic texts in English based on state search; 2) Statistical Machine Translation (SMT). Greene *et al.* (2010) propose a translation model to generation cross-lingual poetry, from Italian to English; 3) Rule-based Templates. Oliveira (2009; 2012) has proposed a system of poem generation platform based on semantic and grammar templates in Spanish. An interactive system has been proposed to reproduce the traditional Japanese poem named *Haiku* based on rule-based phrase search related to user queries (Tosa *et al.*, 2008; Wu *et al.*, 2009). Netzer *et al.* (2009) propose another way of *Haiku* generation using word association rules.

As to computer-assisted Chinese poetry generation. There are now several Chinese poetry generators available. The system named *Daoxiang*¹ basically relies on manual pattern selection. The system maintains a list of manually created terms related to pre-defined keywords, and inserts terms *randomly* into the selected template as a poem. The system is simple but random term selection leads to unnatural sentences.

¹<http://www.poeming.com/web/index.htm>

Zhou *et al.* (2010) use a genetic algorithm for Chinese poetry generation by tonal codings and state search. He *et al.* (2012) extend the couplet machine translation paradigm (Jiang and Zhou, 2008) from a 2-line couplet to a 4-line poem by giving previous sentences sequentially, considering structural templates. Yan *et al.* (2013; 2016) proposed a summarization framework to generate poems. Recently, along with the prosperity of neural networks, a recurrent neural network based language generation is proposed (Zhang and Lapata, 2014): the generation is more or less a translation process. Given previous sentences, the system generates the next sentence of the poem.

We also briefly introduce deep neural networks, which contribute great improvements in NLP. A series of neural models are proposed, such as convolutional neural networks (CNN) (Kalchbrenner *et al.*, 2014) and recurrent neural networks (RNN) (Mikolov *et al.*, 2010) with or without gated recurrent units (GRU) (Cho *et al.*, 2014) and long-short term memory (LSTM) units (Hochreiter and Schmidhuber, 1997). We conduct a pilot study to design neural network structures for couplet generation problems. For the first time, we propose a polishing schema for the couplet generation process, and combine it with the attention mechanism to satisfy the couplet constraints, which is novel.

3 Overview

The basic idea of the Chinese couplet generation is to build a hidden representation of the antecedent clause, and then generate the subsequent clause accordingly, shown in Figure 2. In this way, our system works in an encoding-decoding manner. The units of couplet generation are characters.

Problem formulation. We define the following

formulations:

- *Input.* Given the antecedent clause $\mathcal{A} = \{x_1, x_2, \dots, x_m\}$, $x_i \in \mathcal{V}$, where x_i is a character and \mathcal{V} is the vocabulary, we then learn an abstract representation of the antecedent clause \mathcal{A} .

- *Output.* We generate a subsequent clause $\mathcal{S} = \{y_1, y_2, \dots, y_m\}$ according to \mathcal{A} , which indicates semantic coherence. We have $y_i \in \mathcal{V}$. To be more specific, each character y_i in \mathcal{S} is coordinated with the corresponding character x_i in \mathcal{A} , which is determined by the couplet constraint.

As mentioned, we encode the input clause as a hidden vector, and then decode the vector into an output clause so that the two clauses are actually a pair of couplets. Since we have special characteristics for couplet generation, we propose different neural models for different concerns. The proposed models are extended incrementally so that the final model would be able to tackle complicated issues for couplet generation. We first introduce these neural models from a high level description, and then elaborate them in more details.

Sequential Couplet Generation. The model accepts the input clause. We use a recurrent neural network (RNN) over characters to capture the meaning of the clause. Thus we obtain a single vector which represents the antecedent clause. We then use another RNN to decode the input vector into the subsequent clause by the character-wise generation. Basically, the process is a sequence-to-sequence generation via encoding and decoding, which is based on the global level of the clause. We show the diagram of sequential couplet generation in Figure 2(a).

Couplet Generation with Attention. There is a special phenomenon within a pair of couplets: the characters from the same position in the antecedent clause and subsequent clause, i.e., x_i and y_i , generally have some sort of relationships such as “coupling” or “pairing”. Hence we ought to model such one-to-one correlation between x_i and y_i in the neural model for couplet generation. Recently, the attention mechanism is proposed to allow the decoder to dynamically select and linearly combine different parts of the input sequence with different weights. Basically, the attention mechanism models the alignment between positions between inputs and outputs, so it can be viewed as a local matching model. Moreover, the tonal coding issue can also be addressed by the pairwise attention mechanism. The extension of attention mech-

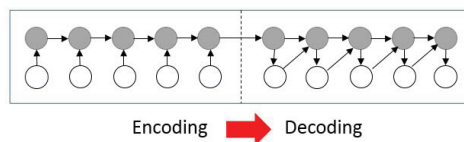


Figure 3: Couplet generation via sequential language modeling: plain neural couplet machine.

anism to the sequential couplet generation model is shown in Figure 2(b).

Polishing Schema for Generation. Couplet generation is a form of art, and art usually requires polishing. Unlike the traditional single-pass generation in previous neural models, our proposed couplet generator will be able to polish the generated couplets for one or more iterations to refine the wordings. The model is essentially the same as the sequential generation with attention except that the information representation of the previous generated clause draft will be again utilized as an input, serving as *additional* information for semantic coherence. The principle is illustrated in Figure 2(c): the generated draft from the previous iteration will be incorporated into the hidden state which generates the polished couplet pair in the next iteration.

To sum up, we introduce three neural models for Chinese couplet generation. Each revised model targets at tackling an issue for couplet generation so that the system could try to imitate a human couplet generator. We further elaborate these neural models incrementally in details in Section 4.

4 Neural Generation Models

4.1 Sequential Couplet Generation

The sequential couplet generation model is basically a sequence-to-sequence generation fashion (Sutskever et al., 2014) using encoding and decoding shown in Figure 3. We use a recurrent neural network (RNN) to iteratively pick up information over the character sequence x_1, x_2, \dots, x_m of the input antecedent clause \mathcal{A} . All characters are vectorized using their embeddings (Mikolov et al., 2013). For each character, the RNN allocates a hidden state s_i , which is dependent on the current character’s embedding x_i and the previous state s_{i-1} . Since usually each clause in the couplet pair would not be quite long, it is sufficient to use a vanilla RNN with basic interactions.

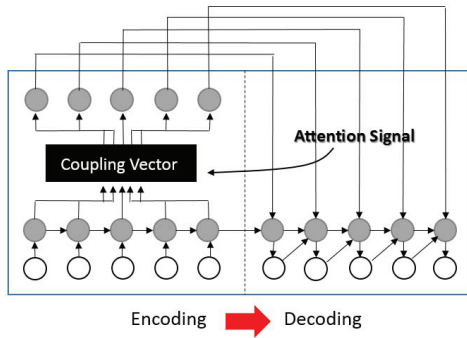


Figure 4: Couplet generation with attention mechanism, namely attention neural coupling machine. Attention signal is generated by both encoder and decoder, and then fed into the coupling vector. Calculation details are elaborated in Section 4.2.

The equation for encoding is as follows:

$$s_i = f(W_h s_{i-1} + W_x x_i + b) \quad (1)$$

x is the vector representation (i.e., embedding) of the character. W and b are parameters for weights and bias. $f(\cdot)$ is the non-linear activation function and we use ReLU (Nair and Hinton, 2010) in this paper. As for the hidden state h_i in the decoding RNN, we have:

$$h_i = f(W_x x_{i-1} + W_h h_{i-1}) \quad (2)$$

4.2 Couplet Generation with Attention

As mentioned, there is special phenomenon in the couplet pair that the characters from the same position in the antecedent clause and the subsequent clause comply with certain relatedness, so that two clauses may, to some extent, look “symmetric”.

Hence we introduce the attention mechanism into the couplet generation model. The attention mechanism coordinates, either statically or dynamically, different positions of the input sequence (Shang et al., 2015). To this end, we introduce a hidden coupling vector $c_i = \sum_{j=1}^m \alpha_{ij} s_j$. The coupling vectors linearly combine all parts from the antecedent clause, and determine which part should be utilized to generate the characters in the subsequent clause. The attention signal α_{ij} can be calculated as $\alpha_{ij} = \sigma_{att}(s_j, h_{i-1})$ after a softmax function. The score is based on how well the inputs from position j and the output at position i match. $\sigma_{att}(\cdot)$ is parametrized as a neural network which is jointly trained with all the other components (Bahdanau et al., 2015; Hermann et al., 2015). This mechanism enjoys the advantage

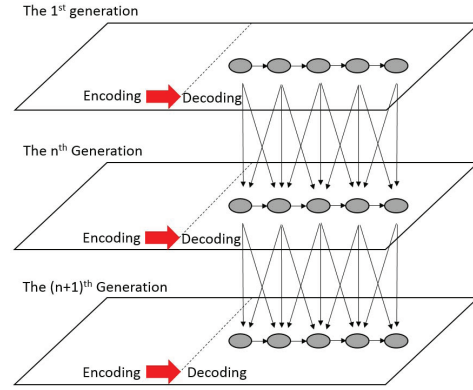


Figure 5: Couplet generation with the polishing schema, i.e., full neural couplet machine. Note that for conciseness, we only show the gist of this schema across polishing iterations. The shaded circles are the hidden vectors to generate characters in the subsequent clause. We omit the duplicated sequential and attention dependencies within each iteration as we have shown in Figures 3 & 4.

of adaptively focusing on the corresponding characters of the input text according to the generated characters in the subsequent clause. The mechanism is pictorially shown in Figure 4.

With the coupling vectors generated, we have the following equation for the decoding process with attention mechanism:

$$h_i = f(W_x x_{i-1} + W_h h_{i-1} + W_c c_i) \quad (3)$$

4.3 Polishing Schema for Generation

Inspired by the observation that a human couplet generator might recompose the clause for several times, we propose a polishing schema for the couplet generation. Specifically, after a single-pass generation, the couplet generator itself shall be aware of the generated clause as a *draft*, so that polishing each and every character of the clause becomes possible.

We hereby propose a convolutionary neural network (CNN) based polishing schema shown in Figure 5. The intuition for convolutionary structure is that this polishing schema guarantees better coherence: with the batch of neighboring characters, the couplet generator knows which character to generate during the revision process.

A convolutional neural network applies a fixed-size window to extract local (neighboring) patterns of successive characters. Suppose the window is of size t , the detected features at a certain position

x_i, \dots, x_{i+t-1} is given by

$$\mathbf{o}_i^{(n)} = f(W[\mathbf{h}_i^{(n)}; \dots; \mathbf{h}_{i+t-1}^{(n)}] + \mathbf{b}) \quad (4)$$

Here $\mathbf{h}^{(n)}$ with the superscript n is the hidden vector representation from the n -th iteration. W and \mathbf{b} are parameters for convolution. Semicolons refer to column vector concatenation. Also, $f(\cdot)$ is the non-linear activation function and we use ReLU (Nair and Hinton, 2010) as well. Note that we pad zero at the end of the term if a character does not have enough following characters to fill the slots in the convolution window. In this way, we obtain a set of detected features. Then a max-pooling layer aggregates information over different characters into a fixed-size vector.

Now the couplet generation with both attention mechanism and polishing schema becomes:

$$\mathbf{h}_i^{(n+1)} = f(W_x \mathbf{x}_{i-1} + W_h \mathbf{h}_{i-1}^{(n+1)} + W_c \mathbf{c}_i^{(n+1)} + W_o \mathbf{o}_i^{(n)}) \quad (5)$$

Note that in this way, we feed the information from the n -th generation iteration into the $(n+1)$ -th polishing iteration. For the iterations, we have the stopping criteria as follows.

- After each iteration process, we have the subsequent clause generated; we encode the clause as \mathbf{h} using the RNN encoder using the calculation shown in Equation (1). We stop the algorithm iteration when the cosine similarity between the two $\mathbf{h}^{(n+1)}$ and $\mathbf{h}^{(n)}$ from two successive iterations exceeds a threshold Δ ($\Delta = 0.5$ in this study).

- Ideally, we shall let the algorithm converge by itself. There will always be some long-tail cases. To be practical, it is necessary to apply a termination schedule when the generator polishes for many times. We stop the couplet generator after a fixed number of recomposition. Here we empirically set the threshold as 5 times of polishing, which means 6 iterations in all.

5 Experiments and Evaluations

5.1 Experimental Setups

Datasets. A large Chinese couplet corpus is necessary to learn the model for couplet generation. There is, however, no large-sized pure couplet collection available (Jiang and Zhou, 2008). As mentioned, generally people regard Chinese couplets as a reduced form of Chinese poetry and there are several large Chinese poem datasets publicly

Table 1: Detailed information of the datasets. Each pair of couplets consist of two clauses.

	#Pairs	#Character
TANG Poem	26,833	6,358
SONG Poem	11,324	3,629
Couplet Forum	46,959	8,826

available, such as *Poems of Tang Dynasty* (i.e., Tang Poem) and *Poems of Song Dynasty* (i.e., Song Poem). It becomes a widely acceptable approximation to mine couplets out of existing poems, even though poems are not specifically intended for couplets² (Jiang and Zhou, 2008; Yan et al., 2013; He et al., 2012). We are able to mine such sentence pairs out of the poems and filtering those do not conform to couplet constraints, which is a similar process mentioned in (Jiang and Zhou, 2008). Moreover, we also crawl couplets from *couplet forums* where couplet fans discuss, practice and show couplet works. We performed standard Chinese segmentation into characters.

In all, we collect 85,116 couplets. We randomly choose 2,000 couplets for validation and 1,000 couplets for testing, other non-overlap ones for training. The details are shown in Table 1.

Hyperparameters and Setups. Word embeddings (Mikolov et al., 2013) are a standard apparatus in neural network-based text processing. A word is mapped to a low dimensional, real-valued vector. This process, known as vectorization, captures some underlying meanings. Given enough data, usage, and context, word embeddings can make highly accurate guesses about the meaning of a particular word. Embeddings can equivalently be viewed that a word is first represented as a one-hot vector and multiplied by a look-up table (Mikolov et al., 2013). In our model, we first vectorize all words using their embeddings. Here we used 128-dimensional word embeddings through vectorization, and they were initialized randomly and learned during training. We set the width of convolution filters as 3. The above parameters were chosen empirically.

Training. The objective for training is the cross entropy errors of the predicted character distribution and the actual character distribution in our

²For instance, in the 4-sentence poetry (namely *quatrain*, i.e., 绝句 in Chinese), the 3rd and 4th sentences are usually paired; in the 8-sentence poetry (namely *regulated verse*, i.e., 律诗 in Chinese), the 3rd-4th and 5th-6th sentences are generally form pairs which satisfy couplet constraints.

corpus. An ℓ_2 regularization term is also added to the objective. The model is trained with back propagation through time with the length being the time step. The objective is minimized by stochastic gradient descent with shuffled mini-batches (with a mini-batch size of 100) for optimization. During training, the cross entropy error of the output is back-propagated through all hidden layers. Initial learning rate was set to 0.8, and a multiplicative learning rate decay was applied. We used the validation set for early stopping. In practice, the training converges after a few epochs.

5.2 Evaluation Metrics

It is generally difficult to judge the effect of couplets generated by computers. We propose to evaluate results from 3 different evaluation metrics.

Perplexity. For most of the language generation research, language perplexity is a sanity check. Our first set of experiments involved intrinsic evaluation of the “perplexity” evaluation for the generated couplets. Perplexity is actually an entropy based evaluation. In this sense, the lower perplexity for the couplets generated, the better performance in purity for the generations, and the couplets are likely to be good. m denotes the length.

$$\text{pow} \left[2, -\frac{1}{m} \sum_{i=1}^m \log p(y_i) \right]$$

BLEU. The Bilingual Evaluation Understudy (BLEU) score-based evaluation is usually used for machine translation (Papineni et al., 2002): given the reference translation(s), the algorithm evaluates the quality of text which has been machine-translated from the reference translation as ground truth. We adapt the BLEU evaluation under the couplet generation scenario. Take a couplet from the dataset, we generate the computer authored subsequent clause given the antecedent clause, and compare it with the original subsequent clause written by humans. There is a concern for such an evaluation metric is that BLEU score can only reflect the partial capability of the models; there is (for most cases) only one ground truth for the generated couplet but actually there are more than one appropriate ways to generate a well-written couplet. The merit of BLEU evaluation is to examine how likely to approximate the computer generated couplet towards human authored ones.

Human Evaluation. We also include human judgments from 13 evaluators who are graduate students majoring in Chinese literature. Evaluators

are requested to express an opinion over the automatically generated couplets. A clear criterion is necessary for human evaluation. We use the evaluation standards discussed in (Wang, 2002; Jiang and Zhou, 2008; He et al., 2012; Yan et al., 2013; Zhang and Lapata, 2014): “syntactic” and “semantic” satisfaction. For the syntactic side, evaluators consider whether the subsequent clauses conform the length restriction and word pairing between the two clauses. For a higher level of semantic side, evaluators then consider whether the two clauses are semantically meaningful and coherent. Evaluators assign 0-1 scores for both syntactic and semantic criteria (‘0’-no, ‘1’-yes). The evaluation process is conducted as a blind-review³

5.3 Algorithms for Comparisons

We implemented several generation methods as baselines. For fairness, we conduct the same pre-generation process to all algorithms.

Standard SMT. We adapt the standard phrase-based statistical machine translation method (Koehn et al., 2003) for the couplet task, which regards the antecedent clause as the source language and the subsequent clause as the target language.

Couplet SMT. Based on SMT techniques, a phrase-based SMT system for Chinese couplet generation is proposed in (Jiang and Zhou, 2008), which incorporates extensive couplet-specific character filtering and re-rankings.

LSTM-RNN. We also include a sequence-to-sequence LSTM-RNN (Sutskever et al., 2014). LSTM-RNN is basically a RNN using the LSTM units, which consists of memory cells in order to store information for extended periods of time. For generation, we first use an LSTM-RNN to encode the given antecedent sequence to a vector space, and then use another LSTM-RNN to decode the vector into the output sequence.

Since Chinese couplet generation can be viewed as a reduced form of Chinese poetry generation, we also include some approaches designed for poetry generation as baselines.

iPoet. Given the antecedent clause, the *iPoet* method first retrieves relevant couplets from the

³We understand that acceptability is a gradable concept, especially for the less subjective tasks. Here from our experience, to grade the “yes”-“no” acceptability is more feasible for the human evaluators to judge (with good agreement). As to couplet evaluation, it might be more difficult for the evaluators to say “very acceptable” or “less acceptable”. We will try to make scale-based evaluation as the future work.

Table 2: Overall performance comparison against baselines.

Algorithm	Perplexity	BLEU	Human Evaluation		
			Syntactic	Semantic	Overall
Standard SMT (Koehn et al., 2003)	128	21.68	0.563	0.248	0.811
Couplet SMT (Jiang and Zhou, 2008)	97	28.71	0.916	0.503	1.419
LSTM-RNN (Sutskever et al., 2014)	85	24.23	0.648	0.233	0.881
iPoet (Yan et al., 2013)	143	13.77	0.228	0.435	0.663
Poetry SMT (He et al., 2012)	121	23.11	0.802	0.516	1.318
RNNPG (Zhang and Lapata, 2014)	99	25.83	0.853	0.600	1.453
Neural Couplet Machine (NCM)	68	32.62	0.925	0.631	1.556

corpus, and then summarizes the retrieved couplets into a single clause based on a generative summarization framework (Yan et al., 2013).

Poetry SMT. He et al. (2012) extend the couplet SMT method into a poetry-oriented SMT approach, with different focus and different filtering for different applications from Couplet SMT.

RNNPG. The RNN-based poem generator (RNNPG) is proposed to generate a poem (Zhang and Lapata, 2014), and we adapt it into the couplet generation scenario. Given the antecedent clause, the subsequent clause is generated through the standard RNN process with contextual convolutions of the given antecedent clause.

Neural Couplet Machine (NCM). We propose the neural generation model particularly for couplets. Basically we have the RNN based encoding-decoding process with attention mechanism and polishing schema. We demonstrate with the best performance of all NCM variants proposed here.

5.4 Performance

In Table 2 we show the overall performance of our proposed NCM system compared with strong competing methods as described above. We see that, for perplexity, BLEU and human judgments, our system outperforms other baseline models.

The standard SMT method manipulates characters according to the dataset by standard translation but ignores all couplet characteristics in the model. The Couplet SMT especially established for couplet generation performs much better than the general SMT method since it incorporates several filtering with couplet constraints. As a strongly competitive baseline of the neural model LSTM-RNN, the perplexity performance gets boosted in the generation process, which indicates that neural models show strong ability for language generation. However, there is a major draw-

back that LSTM-RNN does not explicitly model the couplet constraints such as length restrictions and so on for couplet pairs. LSTM-RNN is not really a couplet-driven generation method and might not capture the corresponding patterns between the antecedent clause and subsequent clause well enough to get a high BLEU score.

For the group of algorithms originally proposed for poetry generation, we have summarization-based poetry method *iPoet*, translation-based poetry method *Poetry SMT* and a neural network based method *RNNPG*. In general, the summarization based poetry method *iPoet* does not perform well in either perplexity or BLEU evaluation: summarization is not an intuitive way to model and capture the pairwise relationship between the antecedent and subsequent clause within the couplet pair. Poetry SMT performs better, indicating the translation-based solution makes more sense for couplet generation than summarization methods. RNNPG is a strong baseline which applies both neural network structures, while the insufficiency lies in the lack of couplet-oriented constraints during the generation process. Note that all poetry-oriented methods show worse performance than the couplet SMT method, indicating that couplet constraints should be specially addressed.

We hence introduce the neural couplet machine based on neural network structures specially designed for couplet generation. We incorporate attention mechanism and polishing schema into the generation process. The attention mechanism strengthens the coupling characteristics between the antecedent and subsequent clause and the polishing schema enables the system to revise and refine the generated couplets, which leads to better performance in experimental evaluations.

For evaluations, the perplexity scores and BLEU scores show some consistency. Besides, we

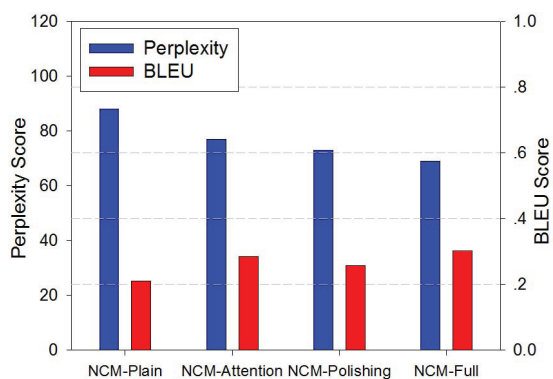


Figure 6: Performance comparison of all variants in the neural couplet machine family.

observe that the BLEU scores are quite low for almost all methods. It is not surprising that these methods are not likely to generate the exactly same couplets as the ground truth, since that is not how the objective function works. BLEU can only partially calibrate the capability of couplet generation because there are many ways to create couplets which do not look like the ground truth but also make sense to people. Although quite subjective, the *human evaluations* in Table 2 can to some extent show the potentials of all couplet generators.

5.5 Analysis and Discussions

There are two special strategies in the proposed neural model for couplet generation: 1) attention mechanism and 2) polishing schema. We hence analyze the separate contributions of the two components in all the neural couplet machine variants. We have the *NCM-Plain* model with no attention or polishing strategy. We incrementally add the attention mechanism as *NCM-Attention*, and then add the polishing schema as *NCM-Full*. The three NCM variants correspond to the three models proposed in this paper. Besides, for a complete comparison, we also include the plain NCM integrated with polishing schema but without attention mechanism, namely *NCM-Polishing*.

The results are shown in Figure 6. We can see that *NCM-Plain* shows the weakest performance, with no strategy tailored for couplet generation. An interesting phenomenon is that *NCM-Attention* has better performance in BLEU score while *NCM-Polishing* performs better in terms of perplexity. We conclude that attention mechanism captures the pairing patterns between the two clauses, and the polishing schema enables better wordings of semantic coherence in the couplet af-

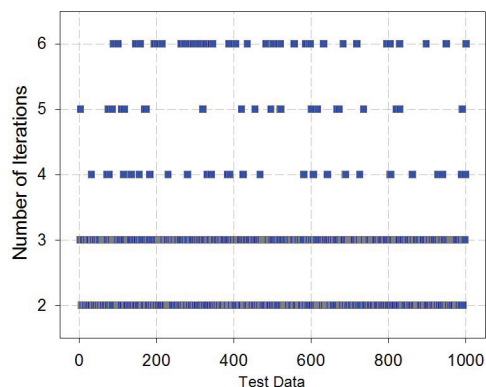


Figure 7: The distribution of stopping iteration counts for all test data. Note that 6 iterations of generation means 5 times of polishing.

ter several revisions. The two strategies address different concerns for couplet generation, hence *NCM-Full* performs best.

We also take a closer look at the polishing schema proposed in this paper, which enables a multi-pass generation. The couplet generator can generate a subsequent clause utilizing additional information from the generated subsequent clause from the last iteration. It is a novel insight against previous methods. The effect and benefits of the polishing schema is demonstrated in Figure 6. We also examine the stopping criteria, shown in Figure 7. In general, most of the polishing process stops after 2-3 iterations.

6 Conclusions

The Chinese couplet generation is a difficult task in the field of natural language generation. We propose a novel neural couplet machine to tackle this problem based on neural network structures. Given an antecedent clause, we generate a subsequent clause to create a couplet pair using a sequential generation process. The two innovative insights are that 1) we adapt the attention mechanism for the couplet coupling constraint, and 2) we propose a novel polishing schema to refine the generated couplets using additional information.

We compare our approach with several baselines. We apply perplexity and BLEU to evaluate the performance of couplet generation as well as human judgments. We demonstrate that the neural couplet machine can generate rather good couplets and outperform baselines. Besides, both attention mechanism and polishing schema contribute to the better performance of the proposed approach.

Acknowledgments

We thank all the anonymous reviewers for their valuable and constructive comments. This paper is partially supported by the National Natural Science Foundation of China (NSFC Grant Numbers 61272343, 61472006), the Doctoral Program of Higher Education of China (Grant No. 20130001110032) as well as the National Basic Research Program (973 Program No. 2014CB340405).

References

- Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*.
- Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*.
- Erica Greene, Tugba Bodrumlu, and Kevin Knight. 2010. Automatic analysis of rhythmic poetry with applications to generation and translation. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, EMNLP'10*, pages 524–533.
- Jing He, Ming Zhou, and Long Jiang. 2012. Generating chinese classical poems with statistical machine translation models. In *Twenty-Sixth AAAI Conference on Artificial Intelligence*, pages 1650–1656.
- Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1684–1692.
- Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation*, 9(8):1735–1780.
- Long Jiang and Ming Zhou. 2008. Generating chinese couplets using a statistical mt approach. In *Proceedings of the 22nd International Conference on Computational Linguistics - Volume 1, COLING '08*, pages 377–384.
- Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.
- Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics on Human Language Technology-Volume 1*, pages 48–54. Association for Computational Linguistics.
- R. Manurung, G. Ritchie, and H. Thompson. 2011. Using genetic algorithms to create meaningful poetic text. *Journal of Experimental & Theoretical Artificial Intelligence*, 24(1):43–64.
- H. Manurung. 2004. An evolutionary algorithm approach to poetry generation. *University of Edinburgh. College of Science and Engineering. School of Informatics*.
- Tomas Mikolov, Martin Karafiát, Lukas Burget, Jan Cernocký, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *INTERSPEECH*, volume 2, page 3.
- Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.
- Vinod Nair and Geoffrey E Hinton. 2010. Rectified linear units improve restricted boltzmann machines. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 807–814.
- Yael Netzer, David Gabay, Yoav Goldberg, and Michael Elhadad. 2009. Gaiku: generating haiku with word associations norms. In *Proceedings of the Workshop on Computational Approaches to Linguistic Creativity, CALC '09*, pages 32–39.
- H. Oliveira. 2009. Automatic generation of poetry: an overview. *Universidade de Coimbra*.
- H.G. Oliveira. 2012. Poetryme: a versatile platform for poetry generation. *Computational Creativity, Concept Invention, and General Intelligence*, 1:21.
- Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.
- Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, ACL-IJCNLP'15*, pages 1577–1586.
- Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pages 3104–3112.
- N. Tosa, H. Obara, and M. Minoh. 2008. Hitch haiku: An interactive supporting system for composing haiku poem. *Entertainment Computing-ICEC 2008*, pages 209–216.

- Li Wang. 2002. A summary of rhyming constraints of chinese poems. Beijing Press.
- X. Wu, N. Tosa, and R. Nakatsu. 2009. New hitch haiku: An interactive renku poem composition supporting tool applied for sightseeing navigation system. *Entertainment Computing-ICEC 2009*, pages 191–196.
- Rui Yan, Han Jiang, Mirella Lapata, Shou-De Lin, Xueqiang Lv, and Xiaoming Li. 2013. i, poet: Automatic chinese poetry composition through a generative summarization framework under constrained optimization. In *Proceedings of the 23rd International Joint Conference on Artificial Intelligence, IJCAI'13*, pages 2197–2203.
- Rui Yan. 2016. i, poet: Automatic poetry composition through recurrent neural networks with iterative polishing schema. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence, IJCAI'16*.
- Xingxing Zhang and Mirella Lapata. 2014. Chinese poetry generation with recurrent neural networks. In *Proceedings of Conference on Empirical Methods in Natural Language Processing*, pages 670–680.
- Kai-Xu Zhang and Mao-Song Sun. 2009. An chinese couplet generation model based on statistics and rules. *Journal of Chinese Information Processing*, 1:017.
- Cheng-Le Zhou, Wei You, and Xiaojun Ding. 2010. Genetic algorithm and its implementation of automatic generation of chinese songci. *Journal of Software*, 21(3):427–437.