International Journal of Knowledge and Language Processing
Volume 3, Number 2, **2012**

www.ijklp.org
KLP International ©2012 ISSN 2191-2734
pp.**1—19**

Topic Model based SongCi Corpus Construction and Research on Computer aided SongCi Writing

HUANG Zixuan[1] and YU Jingsong [2, †]
Department of Language Information Engineering
Peking University
Beijing 100871, China
[1]h123zx567@163.com, [2]yjs@ss.pku.edu.cn

Received March 2012; revised April 2012

ABSTRACT. *SongCi is a typical kind of poetry in China. It is one of the most artistic valuable Chinese ancient literatures. But for the person who wants to write SongCi, it will be a big challenge since a lot of basic knowledge is needed. This paper mainly concerned about how to use computer to help people writing SongCi. To be specific, it means to make use of retrieval capability of computer and known linguistic knowledge to provide reference words to the person who wants to write SongCi. According to the particularity of reference words recommendation, this paper constructs word segmentation and phonology tagging of song ci corpus based on its syntactic features. In order to get the words' styles, this paper adopts the Latent Dirichlet Allocation (LDA) model, which is one of the topic models. By adopting this model, the style analysis task was transferred to topic analysis task. For the collocations extraction, this paper makes proper segmentation based on syntactic features of SongCi, which effectively reduced invalid collocations.*

Keywords: SongCi, Latent Dirichlet Allocation, Computer aided SongCi Writing

1. **Introduction.** In order to write SongCi, except for the knowledge of history and literature which rely on long time works, the following basic knowledge is necessary. Firstly, there are a big number of different tunes for SongCi, each with a typical name, called CiPai; the sentence structures are various among different tunes; one who want to write SongCi should remember well the sentence structures for typical CiPai first. This would be a huge work for an amateur. Secondly, rhyme is another important character for poem-style writing; the number of rhyme words is different according to different reference books, from several thousand to more than ten thousand; without a doubt, to remember these rhyme words is also a tough work. Thirdly, with the limitation of tonal pattern and rhyme scheme, one who wants to express his idea should have a big

† Corresponding author

vocabulary accumulation. If computer can be used as a tool to help people checking tonal patterns, to recommend reference rhyme words, etc., things will be much easier. In addition, by adopting the research results of related fields include analysis of lexical' semantics and topic model, computer can even play an important role in choosing proper lexical collocation and checking words' style.

In this paper, we are trying to find a way to make computer own the ability mentioned above. In order to achieve this, the computer should provide following functions; first one is to provide tonal pattern checking and recommend reference words which correspond to the rules and forms of current position; second one is to provide rhyme words recommendation while a rhyme word is needed at current position; third one is to help people to make up for lacking of vocabulary, that is to say, to recommend words which semantically similar to what the writer want to express. For the first two functions, the main works is to construct related corpus and process automatic retrieve. For the third function, in order to provide reliable recommendation, we considered two ways to achieve this goal. On one hand, we take words collocation as a way to judge priority of reference words. On the other hand, we adopt topic model to create relation between words and the topics which the writers want to express.

This paper is organized as follows. In Section 2, we give a brief review of related works. In Section 3, the circuit of related corpus construction is given. In Section 4, topic analysis of words in SongCi corpus is shown. The design of recommending system is described in Section 5. Section 6 concludes with a summary and suggestions for further works.

2. **Related Works.** Institute of Computational Linguistics of Peking University is one of the institutions which doing related research early in China. In 1980s, the Institute of Computational Linguistics cooperated with Yuan Ze University developed "automatic pinyin-tagging system for the famous Song poems".[1] The system comprehensively considered the tonal pattern of poems, conditional probability, mutual information, and manual rules to tag pinyin for song poems. In addition, they developed a system to help doing research on tang and song poems; constructed tang and song poems' segmentation corpus; after some statistic work, the system provide full-text retrieval of tang and song poems, words based statistical analysis, and sentences similarity retrieval.[2] Besides, Hu JF introduced related methods for computer aided deep research on tang and song poems in his paper.[3] He discussed construction of poem corpus, construction of word list based on statistical method, words segmentation, statistical analysis of lexical semantics, and word-formation rules in poems. In the work of statistical word extraction, three different standards include co-occur frequency, mutual information, and bonding strength were proposed. According to experimental results, combine the three standards would obtain the best results. Among the three standards, the bonding strength is a typical one because it comes from law of word-formation. Therefore, it could make up for the deficiency of statistical methods.

Zhou CL from Xiamen University firstly proposed the concept of “computational poetics” in his book “An Introduction to Computation of Mind and Brain”. He defines “computational poetics” as “using computational ideas, methods and technics to do research work on poems or related literary works”.[4] Institute of artificial intelligence of Xiamen University had done a serials of related research include word segmentation[5], automatic pinyin-tagging[6], and genetic algorithm based automatic generation of Chinese SongCi. [7] Especially, they considered words’ style as a part of fitness function, which is one of the sources of this paper’s idea, although they used a quite simple way to do this work.

Luo FZ from Yuan Ze University began related research at 1993. She mainly concerned about computer aided poem writing and teaching and developed a tonal pattern checking and Chinese classical poem teaching system.[8] The system can provide tonal pattern checking, words retrieve, sentence retrieve and rhyme words retrieve etc. Conclusively, the main work of this system is to construct related database and provide retrieve function. But the most important thing is the idea of using computer to help people writing poems. This paper’s work could be regarded as works follow this idea.

Fei Y from Institute of automation of Chinese Academy of Sciences use artificial neural network to do research on words’ semantics and tested this method by using it on generation of spring festival couplets.[9] Yi Y from Chongqing University considered couplets generation task as a serial generation task in machine learning.[10] He compared several related models include n-gram, hidden markov model, and transformation-based error-driven learning. Besides, Microsoft Research Asia had developed a couplets generation system. They considered the process of couplets generation as the process of translation and adopted some technical from machine translation to do couplets generation[11].

Besides, there are also some funny researches doing by fans on internet, such as high frequency words extraction of SongCi by exhaustion[12], and statistic of relations between writers and Cipais[13]. The methods used in these experiments are simple, but the idea is quite interesting.

3. **SongCi Corpus Construction.**

In order to construct a useful corpus, following works should be done. The first one is oblique tones tagging, which is the foundation of tonal pattern check. The second one is word segmentation, which is the basis of reference words recommendation. The third one is words collocation extraction, which is one of the standards to judge priority of reference words.

3.1. **Oblique Tones Tagging.**

In most previous works, pinyin tagging was done instead of oblique tones tagging. But for our task, tonal pattern checking, oblique tones tagging is more useful than pinyin tagging. Since there are only two different tones in oblique tones other than four in pinyin, it is a comparatively easier work.

We consulted the work of Sui et al.[1], comprehensively considered condition probability and tonal pattern to revise tagging results. The statistical data for condition probability came from “Ancient Chinese Dictionary”.[14] About 24 thousand words included in it, and for each word, the correct pinyin is provided. The tonal pattern data which used to do revise came from “The Authorized Collection of Ci Poem”.[15] This book was written in Qing Dynasty of China, there are 826 CiPai and 2,306 tonal patterns included in it. The pinyin data of Chinese characters came from web, it’s data form as follows.

ID	Character	Pinyin
1	厁	yá ɑes
2	斕	lǎn
3	繫	xì jì
4	纈	xié
5	卸	xiè
6	蒘	yú
7	葱	cōng
8	翉	xié
9	鶋	fú bì
10	蕷	yù
11	闖	chuǎng
12	繼	jì
13	闯	chuǎng
14	倕	chuí
15	麩	fū
16	孄	lǎn
17	膛	tǎng
18	吹	chuī
19	膛	tǎng
20	岚	lǎn
21	薁	yù
22	蕖	xù yù xū

There are 20,872 Chinese characters included in it, and for each character, all of its pronunciation were preserved. We use 0 to indicate “平声”, use 1 to indicate “仄声”, use 2 to indicate “not sure”.

Firstly, we use pinyin data and condition probability data to do initial tagging. For example,

Original Sequence: **归依法，法法不思议。愿我六根常寂静，心如宝月映琉璃。了法更无疑。**

Initial Tagging Sequence: **221 11101 1110011 0011100 11202**

After initial tagging, revise work will be done according to the tonal pattern data. For example,

Initial Tagging Sequence: **221 11101 1110011 0011100 1120**

Tagging Sequence in Tonal Pattern Data: **021 21100 2120011 2001100 01100**

Revised Tagging Sequence: **021 11101 1110011 0011100 11100**

Then, the revised tagging sequence would be considered as the final tagging results.

3.2. **Word Segmentation Corpus Construction.** In order to do word segmentation, a definition of word should be given first. In this paper, we define word as the basic unit the writer used to make up his ideas. In Chinese classical poems, a lot of such kind of units cannot be considered as general words. Take Dufu's poem as an example,

风急天高猿啸哀，渚清沙白鸟飞回。
无边落木萧萧下，不尽长江滚滚来。
万里悲秋常作客，百年多病独登台。
艰难苦恨繁霜鬓，潦倒新停浊酒杯。

Among them, "风急", "沙白", "独登台", "浊酒杯" are not words with general standards. But they are just the basic unit for the writer to express his idea. In practical situation, the usage of these units has no difference from common words. In this paper, according to our definition, we considered these units as words.

For the method of word segmentation, Yu et al. used statistical method to choose characters which tightly bound to each other and highly co-occurrence as word. Then, use these words to build dictionary for further segmentation. [2] Lo FJ used metrical pattern to do word segmentation. Firstly, analysis ling-zi and extract it out. Secondly, for the rest characters, if the number of characters is even, take each two characters as a word, or if the number of characters is odd, extract the last three characters out for further process and take each two characters of the rest as a word. [16] Zhou et al. combined statistical methods with some tonal pattern regulation to do segmentation [5].

According to our definition of word, Lo's method is the most suitable one. Therefore, we choose metrical pattern based methods to do word segmentation.

The key point of metrical pattern based methods is the segmentation of three-character sequence. Because all segmentation task of odd-number character sequence can finally transfer to three-character sequence segmentation task. The work flow of our methods is as follows:

Firstly, we use a word list to do initial segmentation. The word list came from "Ancient Chinese Dictionary" and web resources[17]. For the resolution of overlapping ambiguity, statistical method is used. We comprehensively considered three standards to judge the binding degree of two characters. The first one is mutual information, which is one of the nine common word extraction statistics.[18] The second one is bonding strength of two characters.[2] In addition, frequency of two-character sequences was also taken into account. A voting mechanism is used to integrate all of the three standards and choose the sequence which gets higher score as word.

The formula of mutual information is as follow:

$$I(xy) = \ln \frac{P(xy)}{P_1(x) \times P_2(y)}$$

Among these variables, P1(x) indicates the probability of character “x” occur as the left character in all two-character sequences in corpus; P2(y) indicates the probability of character “y” occur as the right character in all two-character sequences in corpus; P(xy) indicates the probability of character “x” and character “y” occur as two-character sequence “xy” in corpus. Only use mutual information as the standard to judge whether a two-character sequences is a word is not proper. Because mutual information between low frequency characters is often higher than high frequency characters, some other standards are required as a make-up.

The description of bonding strength is as follows. “If two characters can construct a word, when they occur in the same sentence, they tended to occur next to each other.”[3] Define “M” indicates the times of two characters occur next to each other; define “W” indicates the times of two characters occur in the same sentence. Then the formula of bonding strength between the two characters is as follow:

$$D = \left(\frac{M}{W}\right)^2 * \ln(M)$$

Then, comprehensively consider the three standards, “L(xy)”, “D”, “M”, to resolve the overlapping ambiguity.

We construct Ling-zi corpus to resolve the problem of Ling-zi. It includes one-character, two-character and three-character Ling-zi in common use.

又 看 正 乍 恰 奈 似 倘 问 甚 纵 信 便 方 但 料 早 岂 已 漫 凭 有 更 争 怎 任 待
总 向 似 莫 算 况 快 叹 怕 尽 将 未 应 若 想 忆 喜 恨 恐 惊 怨 怅 惜 望 思 念
愁 爱 把 记 见 听 是 被 趁 逢 净 度 湛 尚 与 昔 引 须 唤 哪知 哪堪 犹是 正是 又
是 还见 更是 多少 不须 谁料 谁念 谁知 怎知 只有 何处 莫问 却又 恰又 又还 无端 好
似 恰似 绝似 何奈 记曾 试问 唯有 犹记 犹恨 却喜 独有 漫道 怎禁 忘却 纵把 堪羡 那
番 拼把 因念 追念 自念 又恐 最惜 可喜 追思 追想 遥想 遥望 最可人 最可惜 最无端
最难禁 再休提 更何堪 更何须 又怎知 又匆匆 又早是 又怎知 又安知 又何妨 正依稀 正
销凝 试回看 怎奈向 怎禁得 怎似我 怎忘得 都付与 都应是 记当时 记多情 问此情 问此
中 问何如 问向前 但只愁 但从今 但有时 但眼前 但莫遣 到而今 到如今 待从头 便相将
便准拟 便不成 便与君 君不见 君莫问 问古今 谩从今 谩教人 谩相携 笑从来 笑去年 笑
新来 拼负却 倩何人 嗟多少 空负了 便准拟 要安排 至今想 每追念 似这般 莫不是 况而
今 对此间 愿使君 算年年 悄一如 料也应 似笑我 恨此生 叹人间 收拾起 且消受 记多情
当此际 浑未得 且不是 向黄昏 便明朝 谁共吟 想当年

We first use the Ling-zi data to get the sentences which include Ling-zi. Then, check the result manually and finally we get 21,896 sentences which include one-character Ling-zi, 22,089 sentences which include two-character Ling-zi, and 455 sentences which include three-character Ling-zi.

We random selected 3,000 sentences and do word segmentation manually. We use these sentences as test data to judge the effect of our method. F-value was used as the criterion. Its formula is as follow:

$$F = \frac{2PR}{P+R}$$

Among the 3,000 sentences we chose as test data, 1,000 was three-character sentences, 500 from four-character sentences to seven-character sentences. The sentences which include more than seven-character is relatively few, so they are not chose in this experiment. The experiment result is as follow:

SentNum	SentPrecision	WordsF-value	WordPrecision	WordsRecall
3	0.8974	0.8979	0.8979	0.8979
4	0.9851	0.9816	0.9834	0.9798
5	0.7407	0.8184	0.8184	0.8184
6	0.979	0.9824	0.9853	0.9794
7	0.8014	0.8953	0.8953	0.8953
All	0.9083	0.9324	0.9305	0.9342

It can be seen that for most sentences, this method can get quite good results. But the result for five-character sentences is not so good. We find that the sentence structure of five-character is quite flexible. In many cases, the first character which is not a Ling-zi can also became a word all by itself, such as:

羡 南塘 居士

扫 太虚 纡翳

衬 鱼鳞 浪浅

流 不到 天涯

春 由他 送迎

Therefore, we use individual method for five-character sentences; the work flow of our method is as follows:

1) Get unsegged three-character sequence from left to right.

2) Use the method for three-character sequence to do segmentation; put the left part of the results to the already-segged sequence; put the right part of the results back to the unsegged sequence.
3) Check the length of rest sequence, if it smaller than three, put it to already-segged sequence and go to step 4); else, back to step 1).
4) Check the already-segged sequence, if there are two single-character words next to each other, combine them to a two-character word.

Segmentation Example:

Unsegged Sequence: 花羞人面娇
Get three-character sequence: 花羞人
Do Segmentation：花 羞人
Already-segged sequence: 花
Unsegged Sequence: 羞人面娇**（Bigger than three, back to step 1）**
Get three-character sequence: 羞人面
Do Segmentation：羞 人面
Already-segged sequence: 花 羞
Unsegged Sequence: 人面娇**（Equal to three, back to step 1）**
Get three-character sequence: 人面娇
Do Segmentation：人面 娇
Already-segged sequence: 花 羞 人面
Unsegged Sequence: 娇 **（Smaller than three, put it to already-segged sequence)**
Already-segged sequence: 花 羞 人面 娇
(Do step 4)
Final Result: 花羞 人面 娇

After using individual method for five-character sentences, the experiment result is as follows:

SentNum	SentPrecision	WordsF-value	WordPrecision	WordsRecall
3	0.8974	0.8979	0.8979	0.8979
4	0.9851	0.9816	0.9834	0.9798
5	0.8518	0.8982	0.8982	0.8982
6	0.979	0.9824	0.9853	0.9794
7	0.8014	0.8953	0.8953	0.8953
All	0.9083	0.9324	0.9305	0.9342

The precision of whole sentence is about 0.9082, and the F-value of word segmentation is about 0.9324.

3.3. **Words Collocation Corpus Construction.** After the construction of words

segmentation corpus, the next step is to extract words collocation. The most common method is using bi-gram model, to take all adjacent words as collocation. For example:

杏花/零落/昼/阴阴

By using bi-gram model, the extracted collocation are: “杏花-零落”, “零落-昼”, “昼-阴阴”. One can easily find “杏花-零落” and “昼-阴阴” are valid collocations and “零落-昼” is invalid collocation. In fact, most seven-character sentences in SongCi have the four-three structure. It means that the first four characters are often forming a unit and the last three characters are often forming another. If the sentence was firstly separated into two units, the invalid collocation will no longer be extracted. We use the following form to express such kind of segmentation:

{{**杏花**/B **零落**/E }/B {**昼**/B **阴阴**/E }/E }

The character “B” indicates begin of the unit and “E” indicates end of the unit. For three-character sentences and four-character sentences, there is only one way of sentence segmentation after words segmentation. For five-character sentences, there are three kinds of words segmentation, “221”, “212”, and “122”. For the first two kinds of words segmentation, there is only one way of sentence segmentation. For example:

{无计/B {锁/B 征鞍/E }/E }
{闷倚/B {阑干/B 角/E }/E }

For the “122” style segmentation, ling-zi check should be done first. According to whether the first word is ling-zi or not, there are two different way of sentence segmentation. For example:

{说/B {社稷/B 安危/E }/E }
{{一/B 声声/E}/B 更苦/E}

For the seven-character sentences, there are also three kinds of words segmentation, “1222”, “2212”, and “2221”. For last two kinds of segmentation, according to whether the first word is ling-zi or not, there are two different way of sentence segmentation. For example:

{又恐/B {烟波/B {路/B 隔越/E}/E}/E}
{{红萼/B 无言/E}/B {耿/B 相忆/E}/E}

{试问/B {越王/B {歌舞/B 地/E}/E}/E}
{{小桥/B 流水/E}/B {一枝/B 梅/E}/E}

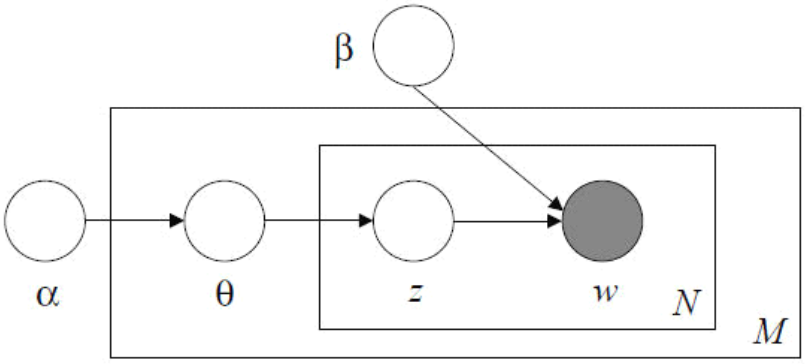
For the “1222” style words segmentation and all six-character sentences, because of its flexibility, we keep use the bi-gram model to extract collocations. For the other sentences which can do sentence segmentation, we only extract the words which belong to the same unit as collocation, for example:

{{铁石/B 心肠/E }/B {为伊/B 折/E }/E }

We only extract “铁石-心肠” and “为伊-折” as collocation. In this way, we can effectively reduce the number of invalid collocations.

4. **Words Topic Analysis for SongCi Corpus.** When a person want to write a SongCi, the exact content he want to express cannot be known by computer, but there must be a subject or topic in his mind. According to the topic in his mind, the words he chooses to express his idea would be with some common point. For example, when a person want to write something about water, he would use words like “悠悠”, “流”, “楼”, “游”, “浮”, “舟”, “ 东流”, etc.; if he wants to describe a beauty, he would like to use words like “翠袖”, “修竹”, “天寒”, “金屋”, “春睡”, “如玉”, “幽独”, “秀色”, etc.. If we can assign each word corresponding topics, and if the writer can provide the topic in his mind, we can restrict the reference words to specific topic and do recommendation under this topic. In this paper, we use Latent dirichlet allocation (LDA) to assign topics to each word.

4.1. **A brief review of LDA.** LDA model is one of the topic models. The aim of topic model is to find hidden topics of large-scale collection of documents. The basic assumption of topic model is to consider a document as a bag of words.[19] It means every word in a document is conditional independent, that is to say, the order of words has no impact to the model. Latent Semantic Indexing (LSI)[20] can be seen as the beginning of topic model. LSI uses the singular value decomposition (SVD) from matrix theory to construct a latent semantics space, and transfer original document to this new space. Although LSI is not a topic model since it is not a probability model, the basic idea of topic model can be seen in LSI. Probabilistic Latent Semantic Index (PLSI)[21] is the first topic model which came from LSI. The idea of PLSI is similar to LSI. But unlike LSI, PLSI use probability model to simulate the process of document generation. After that, Latent dirichlet allocation (LDA) was proposed by Blei et al.[22]. Compare with PLSI, LDA is a more pure probabilistic generation mode. LDA use a k-dimension random variable which obeys dirichlet distribution to indicate topic probability distribution of document. As for PLSI, the probability of document is a parameter to be estimated from model. The graphical representation of LDA is as follows:



The character "β" indicates the probability distribution from topics to words; the character "θ" indicates the multinomial distribution from document to topics and "θ" obeys dirichlet distribution; it's priori parameter is α; the formula is as follows:

$$Dir(\mu \mid \alpha) = \frac{\Gamma(\alpha_0)}{\Gamma(\alpha_1)\cdots\Gamma(\alpha_k)} \prod_{k=1}^{K} \mu_k^{\alpha_k - 1}$$

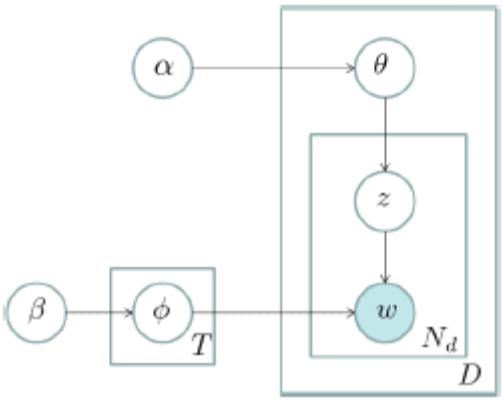
Among these characters:

$$0 \le \mu_k \le 1 \quad , \quad \sum_k \mu_k = 1 \quad , \quad \alpha_0 = \sum_{k=1}^{K} \alpha_k$$

And Γ is the gamma function.
The generation process of a document under LDA model is as follows:

1) Choose N, N obeys Poisson distribution, N indicates the length of document.
2) Choose θ, θ obeys Dirichlet(α) distribution.
3) For each word in N:
   a. Choose topic Zn, Zn obeys multinomial distribution of θ.
   b. Choose Wn accoding to β.

Griffiths et al.[23] added a dirichlet priori parameter to the probability of words to topics. They use β to indicate that parameter and use ϕ to indicate the original β. The graphical representation of the new LDA model is as follows:



The generation process of a document under the new LDA model is as follows:

1) Choose N, N obeys Poisson distribution, N indicates the length of document.
2) Choose θ, θ obeys Dirichlet(α) distribution.
3) For each word in N:
   a. Choose topic Zn, Zn obeys multinomial distribution of θ.
   b. Choose ϕ, ϕ obeys Dirichlet(β) distribution.
   c. Choose word Wn, Wn obeys multinomial distribution of ϕ.

There are several ways for parameter estimation of LDA, include Variational Inference[22], Expectation-Propagation[24], and Gibbs Sampling [25] Each of them has its advantages and disadvantages. In order to choose a proper way, one should comprehensively consider the efficiency, complexity, and accuracy.

4.2. **LDA based Topic Analysis of Words in SongCi Corpus.** According to previous works, there are 23,053 documents in Corpus and the dimension of words is 115,962. We use GibbsLDA++[26] a LDA tool which uses Gibbs Sampling to do parameter estimation, to train our model. The value of α and β are 50/K and 0.01; the iteration number is 1000.

To choose a proper value for K is an important point in LDA model. The value of K will directly influence the topic structure of extracted by the model. There are also several ways to solve this problem. One is using the Hierarchical Dirichlet Process (HDP), which is proposed by Blei et al.[27], to estimate the proper number of K. Another choice is using the PAM model proposed by Li et al.[28], which do not need a previous decided K. Besides, there are also other ways, such as the way proposes by Cao et al.[29].

In this paper, we choose the way propose by Cao et al.. We use the similarity between all topics as the evaluation criterion for choosing value of K. By repeated trial, we can find the value of K which gets the best score as the best value.

The formula to calculate the similarity between two topics is as follow:

$$corre(Z_i, Z_j) = corre(\boldsymbol{\beta}_i, \boldsymbol{\beta}_j) = \frac{\sum_{v=0}^{V} \beta_{iv} \cdot \beta_{jv}}{\sqrt{\sum_{v=0}^{V} (\beta_{iv})^2 \sum_{v=0}^{V} (\beta_{jv})^2}}$$

This means the cosine value of the two vectors of corresponding topics. While the smaller of the cosine value, the more independent between the two topics. Among this formula, β_{iv} indicates the probability from the ith topic to the vth word; β_{jv} indicates the probability from the jth topic to the vth word. Here the β equals to the ϕ in LDA model.

We use the average similarity between all topics to indicate the stability of topic structure. While the smaller of the average similarity, the more stable of the topic structure. The formula is as follow:

$$avg\_corre(structure) = \frac{\sum_{i=1}^{K-1} \sum_{j=i+1}^{K} corre(Z_i, Z_j)}{K \times (K-1)/2}$$

According to different value of K, the average similarity between all topics can be seen from the following curve chart.

We can find that when K is 60, the average similarity reaches its minimum value 0.027615. Hence we choose 60 as the final number of topics.

After training, we will get the following data:

a. Matrix ϕ, as ϕ_{ij} indicates the probability from the i-th topic to j-th word.

b. Matrix θ, as θ_{ij} indicates the probability from the i-th document to j-th topic.

There are two ways to assign topics to words. One is to sort words by the probability from a certain topic to each word and choose the words at the front of the sorted queue; another one is to assign a word the topic which gets the biggest probability from the certain word to each topic.

There advantages and disadvantages for both ways. The first one will ensure each topic get fairly number of words, but will cause some low frequency words getting no topic. The second one will ensure each word get a topic and only one topic. But in fact, there are some words need to be assigned to more than one topic.

According their advantages and disadvantages, we choose a compromising way to assign topics to words. Firstly, we choose the first way to assign words to each topic; in this step, we only choose the word which gets a probability bigger than threshold (we use "Ps" to indicates it). Then, for those words which have not get a topic, we use the second way to assign topic to each of them. According to experiments, we choose 0.0001 as the value of "Ps".

Topic ID	Number of Words	Topic Description
0	8613	多情 一枝 杨柳 不似 瘦
1	8119	新 去年 十分 菊花 日日
2	6268	赋笔 吟笺 迎晓 游处 池亭
3	6148	雪里 疏影 暗香 江梅 因循
4	5644	人物 涌金 风云 无此 障
5	5471	似 芳心 向人 铅华 芳意
6	5363	甚 怀抱 遥想 素秋 旧家
7	4813	勋业 诗书 役 胸中 风涛
8	4662	事业 知否 友 堪怜 野草
9	4591	要 鼓 抚 蛟龙 天意
10	4447	羡 何为 杯酒 君行 君子
11	4154	谩 未许 许大 清绝 淡月
12	3870	几度 眉妩 晋人 深窈 古意
13	2281	极 降 俨 保 奉
14	2074	惨 紫云 无因 告 晓
15	1982	拥 扬州 太守 使君 吐
16	1937	恶 约 薄 角 伤
17	1892	兰亭 五云 曲水 一觞 清都
18	1646	年年 朱颜 蟠桃 千岁 祝
19	1533	惊起 坠 名 横塘 鸳鸯
20	1446	山中 一杯 且 老矣 聊
21	1410	中 月明 心 帘栊 梧桐
22	1405	人生 如何 万事 世间 也
23	1326	园林 天气 清明 寻芳 踏青
24	1255	年 二十 此情 何用 谁共
25	1143	一片 潇湘 相 并 摇

26	1116	青山 细雨 落日 晚来 白蘋
27	1073	一枕 坐 草草 可 相望
28	999	秋风 南北 儿 古今 相忆
29	985	红 歌 翠袖 荐 白雪
30	959	为 漫 东篱 重阳 亦
31	920	说与 霓裳 误 婵娟 边
32	913	江南 年华 杜鹃 尽日 风景
33	890	花间 影里 迟迟 更有 言语
34	889	此时 笑语 步 玉堂 眼
35	848	思 船 娟娟 词 山川
36	806	雨 处 暮 江上 山
37	772	喜 趁 寿 通 人知
38	764	黄花 如此 江山 今古 几番
39	759	三十 眠 去后 没 白头
40	745	登临 倒 拚 天风 朝
41	724	秋 愁 休 万里 悠悠
42	698	故人 扁舟 客 江湖 阳关
43	665	南枝 前村 羌管 留取 昨夜
44	639	芳草 流水 斜阳 回首 垂杨
45	597	凄凉 试 双 都是 陌上
46	591	佳人 月下 先生 幽香 竹
47	582	落 纷纷 江城 此花 仍
48	564	欲 渊明 门 三径 有酒
49	508	相思 寄 苦 诗 离恨
50	455	一笑 无言 足 移 俱
51	434	应 真 何许 夜阑 好事
52	371	又 富贵 何须 悲 长年
53	247	天涯 明朝 酒醒 人家 斜
54	193	分付 古 朝朝 浪 元
55	136	归去 酒 谁能 来兮 出岫
56	131	与 老子 遣 本 不可
57	128	发 乐 月上 樽前 夜半
58	128	不是 黄金 底 只在 镜里
59	126	燕子 楼 帘卷 征鸿 目送

We choose the first five words which get the biggest probability as the description of the topic. The number of words assigned to each topic and the description can be seen in the top table.

We filtered the topics which get too few words and finally get 50 topics.

Now we get all the data we want, oblique tones for each word, words collocation, and topics for each word. The data format is as follow:

ID	Word	TonalPattern	FirstChar	SecondChar	rhyme	Left_Collocation	Right_Collocation
1	旧时	10	旧	时	上平四支	有、比似、却是、歌舞	月、标格、曾在、雨、
2	月色	11	月	色	入声十三职	恐随、今宵、中、女墙	侵、冷如、兼天、波光
3	算	1	算	算	去声十五翰	心计、称寿、祝椿、只	明年、有、万事、何必
4	几番	20	几	番	上平十三元	吹老、销磨、拔帜、衾	风、梦回、花落、南极
5	照我	11	照	我	上声二十哿	几番、十分、又、还、	当楼、登楼、看看、一
6	梅边	0	梅	边	下平一先	早似、柳底、款语、算	只欠、瘦、粉瘦、香沁
7	吹笛	0	吹	笛	入声十二锡	月明、月中、小楼、弹	月明、归去、鱼龙、到
8	唤起	11	唤	起	上声四纸	怕、如何、为、一声、	淡妆、长风、愁、醒松
9	王人	10	王	人	上平十一真	有、且共、偏称、曾伴	劝客、羞懒、更著、低
10	不管	11	不	管	上声十四旱	人、东君、空归、都、	离心、梨花、清寒、轻
11	清寒	0	清	寒	上平十四寒	一路、破、贮、水石、	髓、满袖、瘦、初溢、

4.3. **Experiments for Topic Analysis of Words.** Now we can do word search according to its tonal pattern, first character, last character, rhyme, left collocation, and right collocation.

We choose the Chinese character “紫” as an example to do first character search. First we choose the topic “**佳人 月下 先生 幽香 竹**”. This topic is about beauty. We get the following words:

紫玉 紫腰

Among them, “**紫玉**” is the name of FuChai’s littler daughter. FuChai is a king in Chunqiu Period of ancient China; “**紫腰**” comes from the Ci poem written by ZhaoChangqin; The original sentence is “**紫腰艳艳，青腰袅袅，风月具闲。**”. We can easily find that it was used to describe beautiful girls.

Then we change the topic to“**年年 朱颜 蟠桃 千岁 祝**”. This topic is about blessing in festivals. We get the following words:

紫皇 紫金 紫绶 紫泥 紫府 紫盖

All of these words have the meaning of wealth and luck.

Now we take a look at collocation, using “凭栏” as the search word to do right collocation search. First we choose the topic “**人生 如何 万事 世间 也**”. This topic is about thinking on our life. We get the following collocations:

凭栏-笑 凭栏-幽思

Then we choose the topic“**芳草 流水 斜阳 回首 垂杨**”. This topic is about human’s emotion when they saw some natural phenomena. We get the following collocations:

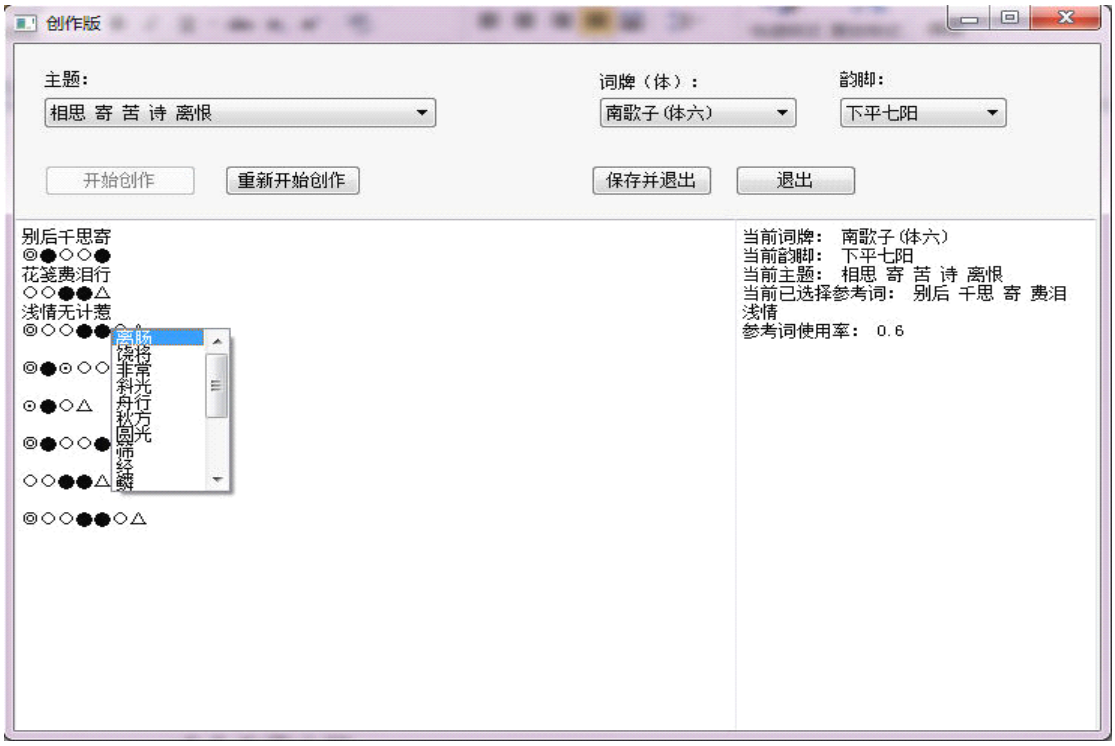
凭栏-无语 凭栏-送目

We can see the words under different topics have their particular features. The LDA model has done a good job.

5. **The design of recommending system.** The work flow of the recommending system is as follows:

1) Get the topic, CiPai, and rhyme from the user.
2) Get the written part of current sentence and do word segmentation.
3) Get the last written word (is there is any)
4) According to the data of CiPai corpus, get current tonal pattern for recommending word; if a rhyme word is needed at current position, add the rhyme as a search parameter.
5) Using the related factors we get, including collocation word, tonal pattern, rhyme, and topic, to do word search and return results to the user.

The following is an example about how the recommending system works.



The use ratio of recommending words is about 0.6. Our system does help the user writing SongCi.

6. **Conclusion and Future Works.** In this paper, we constructed a CongCi corpus based on LDA model and tested topic based reference words recommendation. According to our experiments, we found the LDA model does a good job on topic analysis and the recommending words do make sense.

As a future work, we will consider the task of style analysis of sentences or poems, as well as the style analysis of works of a certain writer, by using topic model. Moreover, we plan to analysis the style distribution of a certain CiPai.

7. **Acknowledgment.** This paper is supported by the Humanities and Social Science Research Projects Fund (No. 10YJC740124) from Ministry of Education of the People's Republic of China.

REFERENCES

[1] Barbara R., Marti A. H., Charles J. F.: The Descent of Hierarchy, and Selection in Relational Semantics. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 247-254, University of Pennsylvania, 2002.Sui, Z.-F., Yu, S.-W., Lo, F.-J. The research on automatic pinyin-tagging for the famous Song poems and its implementation. Journal of Chinese Information Processing, 1998, 12(2):44 53 (in Chinese with English abstract).

[2] Yu, S.-W., Hu, J.-F. Word-Based statistical analysis of Chinese ancient poetry. Language and Linguistics, 2000, 4(3):631 647 (in Chinese with English abstract).

[3] Hu, J.-F. The lexicon meaning analysis-based computer aided research work of Chinese ancient poems [Ph.D. Thesis]. Beijing: Peking University, 2001 (in Chinese with English abstract).

[4] Zhou, C.-L. An Introduction to Computation of Mind and Brain. Beijing: Tsinghua University Press, 2003 (in Chinese).

[5] Su, J.-S., Zhou, C.-L., Li, Y.-H. The establishment of the annotated corpus of Song dynasty poetry based on the statistical word extraction and rules and forms. Journal of Chinese Information Processing, 2007,21(2):52-57 (in Chinese with English abstract).

[6] [6] Lai, X.-B., Zhou, C.-L. The Research on Grapheme-to-Phoneme Conversion for Song Dynasty Poetry and Its Implementation. Cognitive and Computation, 637-643 (in Chinese with English abstract).

[7] Zhou, C.-L., You, W., Ding, X.-J. Genetic algorithm and its implementation of automatic generation of Chinese SONGCI. Journal of Software, 2010,21(3):427 437.

[8] Lo, F.-J., Lee, Y.-P., Tsao, W.-C. The format auto-checking and database indexing teaching system of Chinese poetry and lyrics. Journal of Chinese Information Processing, 1999,13(1):35 42 (in Chinese with English abstract).

[9] Fei, Y. Research on multi-level integration of chinese semantics and system design of spring festival couplets [Ph.D. Thesis]. Beijing: Institute of Automation Chinese Academy of Science, 1999 (in Chinese with English abstract).

[10] Yi, Y. A study on style identification and Chinese couplet responses oriented computer aided poetry composing [Ph.D. Thesis]. Chongqing: Chongqing University, 2005 (in Chinese with English abstract).

[11] Zhou, M. Microsoft's generation system of Chinese couplets. Microsoft Research Asia natural language processing group. Beijing.2006(in Chinese with English abstract). http://duilian.msra.cn/

[12] http://cos.name/2011/03/statistics-in-chinese-song-poem-1/

[13] http://cos.name/2012/03/statistics-in-chinese-song-poem-2/

[14] Ancient Chinese Dictionary. Beijing: The Commercial Press

[15] The Authorized Collection of Ci Poem. Beijing: Chinese Book Store, 1983 (in Chinese).

[16] Lo, F.-J. The design and application of the system for poetic language segmentation and semantic

classification tagging. In: Proc. Of the 4th Symp. on digital reservation. 2005 (in Chinese with English abstract).

[17] http://hanyu.iciba.com/

[18] Lo, S.-F., Sun, M.-S. Chinese Word Extraction Based on the Internal Associative Strength of Character Strings. Journal of Chinese Information Processing, 2003, 17(3):9-14 (in Chinese)

[19] Xu, G., Wang, H.-F. The Development of Topic Models in Natural Language Processing[J]. Chinese Journal of Computers, 2011, 34(8):1423-1434

[20] Deerwester, S.C., Dumais, S.T., Landauer, T.K., et al. Indexing by latent semantic analysis [J]. Journal of the American Society for Information Science, 1990, 41(6):391-407

[21] Hofmann, T. Probabilistic latent semantic indexing. Proceeding of the 22nd Annual International SIGIR Conference, New York: ACM Press,1999:50-57

[22] Blei, D., Ng, A., Jordan, M. Latent Dirichlet allocation [J]. Journal of Machine Learning Research, 2003, 3:993-1022

[23] Griffiths, T.L., Steyvers, M. Finding scientific topics [A]. Proceeding of the National Academy of Sciences [C], 2004, 101:5228-5235

[24] Minka, T., Lafferty, J. Expectation propagation for the generative aspect model [A]. Proceeding of UAI2002 [C], Edmonton, Alberta, Canada, 2002:352-359

[25] Heinrich, G. Parameter estimation for text analysis. http://www.arbylon.net/publications/text-est.pdf

[26] http://gibbslda.sourceforge.net/

[27] The, Y., Jordan, M., Beal, M., Blei, D. Hierarchical dirichlet processes [J]. Journal of the American Statistical Association, 2007,101(476):1566-1581

[28] Li, W., McCallum, A. Nonparametric bayes pachinko allocation [A]. Proceedings of the UAI [C], Vancouver, BC, Cannada, 2007.

[29] Cao, J., Zhang, Y.-D., Li, J.-T., Tang,S. A Method of Adaptively Seleting Best LDA Model Based on Density[J]. Chinese Journal of Computers, 2008, 31(10):1780-1787