# A Pragmatic Approach for Classical Chinese Word Segmentation

**Shilei Huang, Jiangqin Wu**
College of Computer Science and Technology, Zhejiang University
Yuquan Campus, Zhejiang University, Zheda Road, Hangzhou, Zhejiang, China
{21621129, wujq}@ zju.edu.cn

**Abstract**

Word segmentation, a fundamental technology for lots of downstream applications, plays a significant role in Natural Language Processing, especially for those languages without explicit delimiters, like Chinese, Korean, Japanese and etc. Basically, word segmentation for modern Chinese is worked out to a certain extent. Nevertheless, Classical Chinese is largely neglected, mainly owing to its obsoleteness. One of the biggest problems for the researches of Classical Chinese word segmentation (CCWS) is lacking in standard large-scale shareable marked-up corpora, for the fact that the most excellent approaches, solving word segmentation, are based on machine learning or statistical methods which need quality-assured marked-up corpora. In this paper, we propose a pragmatic approach founded on the difference of t-score (dts) and Baidu Baike (the largest Chinese-language encyclopedia like Wikipedia) in order to deal with CCWS without any marked-up corpus. We extract candidate words as well as their corresponding frequency from the *Twenty-Five Histories* (*Twenty-Four Histories* and *Draft History of Qing*) to build a lexicon, and conduct segmentation experiments with it. The F-Score of our approach on the whole evaluation data set is 76.84%. Compared with traditional collocation-based methods, ours makes the segmentation more accurate.

**Keywords:** Classical Chinese, Word Segmentation, Difference of T-score, Encyclopedia, Collocation

## 1. Introduction

Till now, great achievements have been made in modern Chinese word segmentation (CWS) while there is little progress in Classical Chinese word segmentation (CCWS) for the obsoleteness of Classical Chinese. However, it's worth studying it. As a fundamental technology, word segmentation is a prerequisite of making deep analyses on Classical Chinese literature. From the word frequency we can discover the transition of words which can explain some linguistic phenomena, historical facts, traditional folk cultures, social culture, geographical information and etc. Furthermore, we can extract entities like person, location, official title, date and so forth, to dig into the history and culture, rather than get superficial knowledge by basic search like string matching. Additionally, many downstream applications like knowledge graph and QA system need to be built from these entities. In recent years, many researchers have made analyses and explorations of unstructured or structured Classical Chinese literature (Xu, 2016; Ouyang, 2016; Liu et al., 2015; Lee and Wong, 2013; Li et al., 2012; Fang, Lo and Chinn, 2009).

According to the estimate of experts, the number of the existing Chinese ancient books is about from 80,000 to 100,000. More than 40,000 books (around 4.8 billion Chinese Characters) have been converted into digital version (Ouyang, 2016). However, there are few marked-up corpora, let alone some of them being not shareable. Currently, the most known Classical Chinese corpora, which are marked up with POS (part of speech) tags and segmented into words, are Academia Sinica Ancient Chinese Corpus[1], Sheffield Corpus of Chinese[2] and CityU Treebank of Classical Chinese Poems[3]. Unfortunately, they are not enough to do a comprehensive analysis over a long period of history. And for the inconsistency of tagging and segmenting standards of the corpora, it's quite tough to aggregate them.

Building Classical Chinese corpus is much more expensive than building modern one. On the one hand, there are unique words and expressions in each dynasty, some of which only appear within a certain period of time and are no longer used in modern life. The meaning of words is changing along with time. Thus, comprehensive relevant knowledge is required to segment the sentence. On the other hand, the range of the Classical Chinese literature is generally from the end of the Spring and Autumn period through to the end of the Qing dynasty, nearly 3000 years. Only with considerable literature of each era being marked-up, can we make an all-sided analysis on literatures over the long history by means of supervised approaches. Nonetheless, it's costly and inefficient to do so. For instance, it took Shi el al. (2010) 4 years to manually segment and annotate word POS of 25 literatures of Pre-Qin era.

In order to tackle these problems, we propose a pragmatic approach to create a lexicon, with which researchers are able to segment the Classical Chinese texts for other pertinent studies or get preprocessed corpus for creating a more accurate one efficiently. The basic idea behind our method is that firstly, we get the statistical information of all bigrams by iterating texts in the corpus; secondly, when we iterate the texts again, candidate words (collocation) as well as their frequency are extracted in accordance to the thought of integrating the difference of t-score with Baidu Baike.

The contributions of this paper are as follows:

1. To the best of our knowledge, this is the first work that combines encyclopedia and traditional statistical methods to construct lexicon for word segmentation.
2. From what we can tell, this is the first work that studies on CCWS throughout the whole history of Classical Chinese literature.
3. Alleviate the problem of lacking in standard large-scale shareable marked-up corpora of Classical Chinese.
4. Comparison experiments are carried out to prove the effectiveness of our approach in CCWS.

The rest of this paper is organized as follows. Related work will be introduced in Section 2. Data we use will be explained in Section 3. Methodology and experiments will

[1] http://ancientchinese.sinica.edu.tw/c_intro.html

[2] https://hridigital.shef.ac.uk/scc/

[3] http://classicalchinese.lt.cityu.edu.hk/

be introduced in Section 4 and Section 5 respectively. Finally, concluding remarks are given in Section 6.

# 2. Related Work

Up to now, numerous approaches of word segmentation have been proposed. These methods can be roughly classified as lexicon-based, statistically-based or neural network-based methods. Hybrid methods are not introduced here.

## 2.1 Lexicon-based approaches

The most popular lexicon-based algorithms are forward maximum matching method (FMM), and backward maximum matching method (BMM). The performance of these methods mostly depends on the coverage of the lexicon. Qiu and Huang (2008) proposed a heuristic hybrid CCWS method. Using *Hanyu Da Cidian* as a basic dictionary, segments the texts with BMM. For increasing the accuracy, they count the frequency of words that have already appeared, extract the words with high frequency and add new words to the word list. As a general-purpose dictionary, the limitation of *Hanyu Da Cidian* is rather apparent, like sparseness, data incompleteness and bias when it's utilized for CCWS over literature of different eras. Since there is no appropriative dictionary, researchers tend to combine lexicons with statistical methods instead of adopting lexicon-based approaches alone.

## 2.2 Statistically-based approaches

Statistically-based approaches can be further divided into labelling-based or collocation-based (corpus-based) methods.

Xue (2003) first proposed the character-based tagging approach, which treats word segmentation as a sequence tagging problem, assigning corresponding labels to the characters. The labels indicate the location of each character, beginning of, inside or end of a certain word. Xue's work leads to some subsequent researches (Peng, Feng and McCallum, 2004; Tseng et al., 2005) on integrating character labelling with statistical models like HMM, MEMMs, CRF and etc. Afterwards, Zhang and Clark (2007) took the word-level information into consideration, proposing a word-based CWS approach using a discriminative perceptron learning algorithm. The prerequisite of these kinds of solutions is high-quality marked-up corpora, which are hard to acquire. Therefore, at present, solving CCWS for specific ancient books or a specific period of time is the main trend (Shi, Li and Chen, 2010).

In terms of collocation, mutual information is the most used metric. As a concept of information theory, mutual information is used to measure the degree of association between two Chinese characters. The higher the mutual information is, the more related the two characters are. Sproat and Shih (1990) utilized mutual information to quantificationally describe how strongly associated of two arbitrary characters, found upon which bigrams are extracted automatically from raw corpus. Sun, Shen and Benjamin (1998) introduced the difference of t-score (dts) between Chinese characters, proposing an automatic segmentation algorithm based on mutual information and dts. In recent two decades, a number of scholars have focused on Chinese word extraction for solving CWS or CCWS by dint of various association metrics or a hybrid model of several metrics (Chang and Su, 1997; Chen and Ma, 2002; Luo and Sun, 2003; Ma and Chen, 2003; Feng et al., 2004; Tang et al., 2009; Zhang et al., 2009; Zhang et al., 2010; Duan, Han and Song, 2012; Zhang et al., 2012; Mei et al., 2015; Shen, Kawahara and Kurohashi, 2016). However, one big problem for collocation-based methods is that the performance of certain systems will heavily depend on the thresholds setting, because the thresholds for such association metrics are always set heuristically or empirically to gain high performance, which means we have to adjust thresholds with the change of applications or domains with extra effort. The approach we propose in this paper cuts the Gordian knot by integrating encyclopedia with traditional association metrics, with which there is no need setting threshold for deciding whether to discard a potential word or not.

## 2.3 Neural network-based approaches

With the rise of deep learning, neural models have been widely used for NLP tasks to avoid the task-specific feature engineering. Zheng, Chen and Xu (2013) performed CWS and POS tagging by adapting a general neural network architecture for sequence labeling. Pei, Ge and Chang (2014) improved upon Zheng's work by modeling the interactions between local context and previous tag. Chen et al. (2015a) proposed a gated recursive neural network, modeling the feature combinations of context characters. With the purpose of alleviating the limitation of the size of context window, Chen et al. (2015b) utilized a LSTM architecture to capture potential long-distance dependencies. Cai and Zhao (2016) proposed a novel neural framework which thoroughly eliminates context windows and can utilize complete segmentation history.

As far as we know, neural network-based methods have not been used to tackle CCWS yet. From our perspective, the diversity of Classical Chinese literature and the extreme lack of marked-up corpora account for it.

Fortunately, only with a certain size of raw corpus and a certain number of online encyclopedia documents, can we get acceptable result over CCWS. In Section 3 and Section 4, we will explain the corpus as well as other resources we use, and the concrete process how to segment texts with the lexicon we build in detail.

# 3. Corpus and resources

## 3.1 Raw corpus

We choose the *Twenty-Four Histories* and the *Draft History of Qing* as raw corpus, totally containing 3742 volumes and around 31 million characters. They are the Chinese official historical books, covering a period from 3000 BC to the end of Qing Dynasty (1912 AD), considered as one of the most important sources on Chinese history and culture. The books mostly record pertinent activities of emperors and politicians. Additionally, the content covers economy, politics, culture, art, astronomy, law, geography, science, technology and etc.

## 3.2 Evaluation data set

Owing to the fact that there is no standard datasets for CCWS and the performance of our approach can only be fully demonstrated by testing on literature of different eras, we randomly select a certain number (in proportion to the size of each book) of texts from each book and segment the sentences manually. The test data contains 32689 characters in total, covering every aspect of the content.

### 3.3 Encyclopedia documents

10,143,321 documents were crawled from Baidu Baike[4] which is the largest Chinese-language online encyclopedia. For now, there are more than 15 million pages and more than 6 million people get involved in this project. The encyclopedia is a huge external knowledge resource, from which we can fetch concepts and entities. Like what we stated above, Classical Chinese is nearly obsolete, which means new content will not be generated. At the same time, the encyclopedia will gradually be complete in terms of knowledge about Classical Chinese literature. The title of each document can be considered as a word or a combination of words. That's why we attempt to integrate traditional statistical ways with Baidu Baike to extract words from Classical Chinese literature.

What we just need is the title of each entry. However, a dictionary with all the titles will cause low efficiency or memory problem. We remove some of them to shrink the volume. Firstly, the titles, which contain more than 8 or less than 2 characters, or contain non-Chinese character (like digits and punctuations), are gotten rid of. Then, we get 6,610,492 titles left. Secondly, remove those that contain high frequency (greater than 1000) prefix or suffix, some of which are listed in Table 1. The count of distinct affixes within corresponding frequency range is listed in Table 2. Finally, there are 4,324,035 distinct titles left.

| Affixes | Frequency |
|---|---|
| 酒店 | 68480 |
| 中国 | 31220 |
| 中学 | 30550 |
| 穿越 | 19361 |
| 社区 | 18696 |

Table 1: Frequency of some prefix or suffix

| Frequency range | 10-100 | 100-500 | 500-1000 | >1000 |
|---|---|---|---|---|
| Count | 21701 | 8779 | 1540 | 992 |

Table 2: Count of prefix or suffix within a certain range

### 3.4 Official titles from CBDB

We extract Chinese ancient official titles from CBDB[5] (China Biographical Database Project), a project of Harvard University. As supplementary, 21152 distinct official titles that meet the requirements of the above preprocessing, are merged together with titles from Baidu Baike encyclopedia.

## 4. Methodology

Our approach mainly focuses on the creation of lexicon. Not only the vocabulary but also its corresponding frequency in the raw corpus are extracted. With such a lexicon, maximum probability, the commonly used algorithm for word segmentation, is applied to CCWS in our method.

### 4.1 Lexicon creation

The general procedure is illustrated in Figure 1. From the first iteration over texts in the corpus, frequency of each Chinese character bigram is recorded. In the light of Aho-Corasick Automaton, a temporary dictionary is built from the titles extracted from Baidu Baike and CBDB, with which we can rapidly get the location of each word in the sentence. Before explaining the following procedures, we will give the definitions of t-score and dts first.

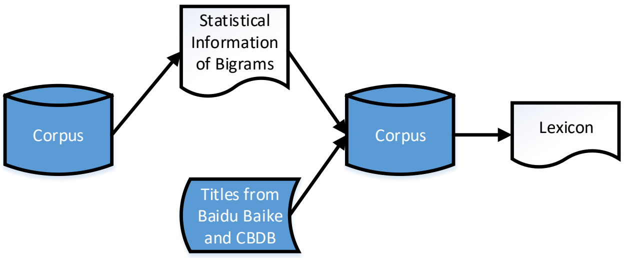
Statistical Information of Bigrams
Corpus
Corpus
Lexicon
Titles from Baidu Baike and CBDB

Figure 1: Flow diagram of lexicon creation.

Given a Chinese character string 'xyz', the t-score of the character y relevant to x and z is defined as:

$$ts_{x,z}(\mathrm{y}) = \frac{p(z|y) - p(y|x)}{\sqrt{var\big(p(z|y)\big) + var\big(p(y|x)\big)}}$$

Where p(y|x) is the conditional probability of y given x, and p(z|y), of z given y, and var(p(y|x)), var(p(z|y)) are variances of p(y|x) and of p(z|y) respectively.

Given a Chinese character string 'vxyw', the dts between characters x and y is defined as:

$$\mathrm{dts}(\mathrm{x}\!:\mathrm{y}) = ts_{v,y}(x) - ts_{x,w}(y)$$

If dts(x:y) > 0 then it tends to be bound. If dts(x:y) < 0 then it tends to be separated. More details about t-score and dts, please refers to the work of Sun et al. (1998) and Church et al. (1991).

The main processes of the second iteration are as follows:

**Step 1:** Segment one sentence into sub-sentences (sub-sentence here refers to a Chinese character string without any punctuation). Then, iterate over the sub-sentences.

**Step 2:** Calculate dts of every bigram of the sub-sentence.

**Step 3:** Look up the dictionary and list all words of this sub-sentence.

**Step 4:** Skip over bigram-words. For words that contain more than 3 characters, add them to lexicon directly (if the word exists, just increase corresponding frequency，similarly hereinafter). For words that exactly contain 3 characters, we need to consider about three situations. If the last character of previous word is same as first character of current word and next word (bigram-word) is same as the last two characters of current word, add previous word and next word to lexicon; if the previous word is same as the first two characters of current word and the first character of next word is same as last character of current word, add previous word and next word to lexicon; otherwise, add current word to lexicon. Overlap is not allowed in this step, which means each character can only belong to a specific word.

**Step 5:** Find out the position with largest dts among the left characters of the sub-sentence. Take the position as center, search bidirectionally, and bind characters together if the dts between two characters is greater than zero. Keep searching until the dts is smaller than zero or no character left. Then, we get a candidate word. If this word contains

[4] https://baike.baidu.com/

[5] https://projects.iq.harvard.edu/cbdb

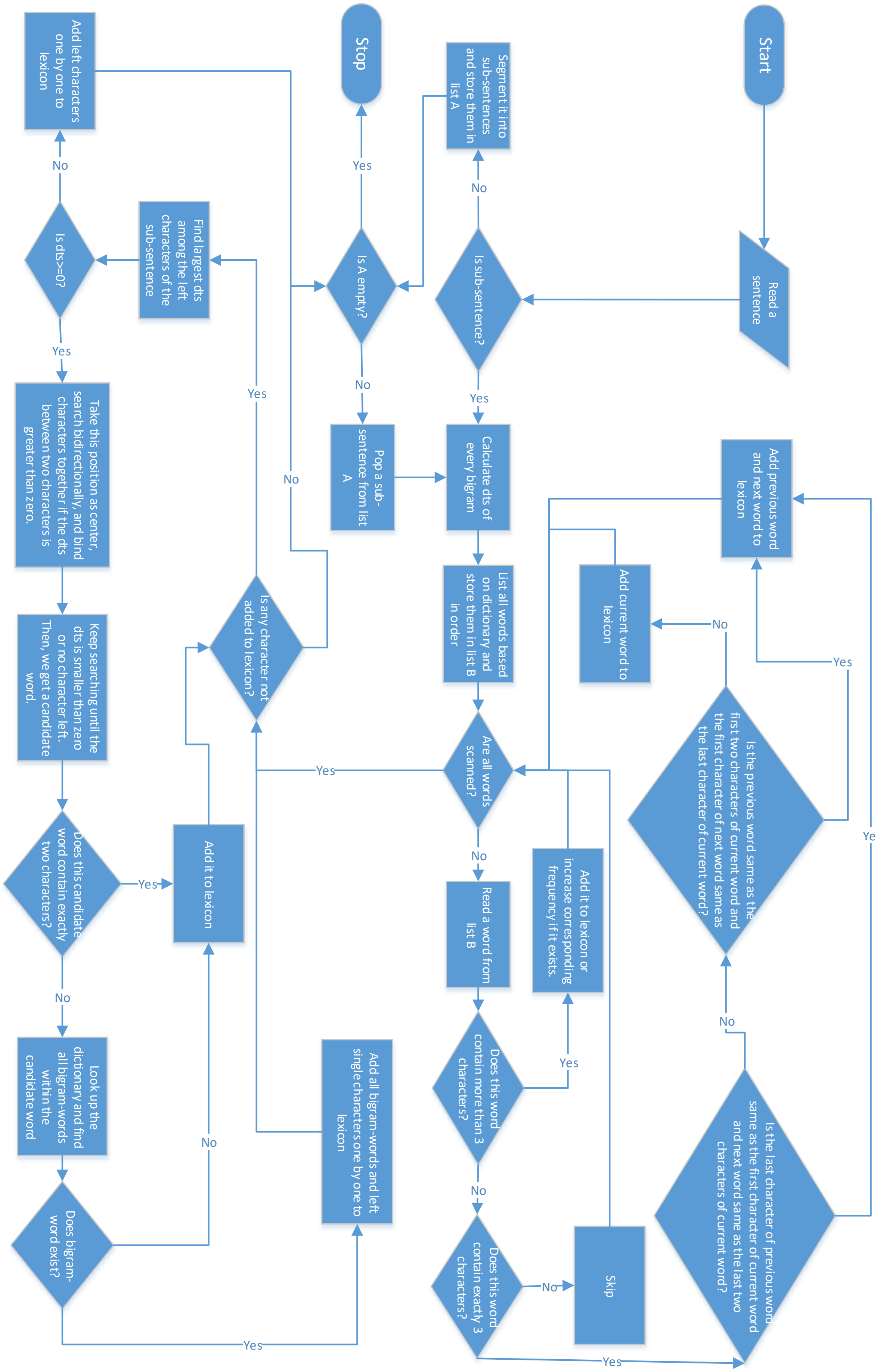
Start
Read a sentence
Is sub-sentence?
No
Segment it into sub-sentences and store them in list A
Yes
Calculate dts of every bigram
Is A empty?
Yes
Stop
No
Pop a sub-sentence from list A
List all words based on dictionary and store them in list B in order
Are all words scanned?
No
Read a word from list B
Does this word contain more than 3 characters?
Yes
Add it to lexicon or increase corresponding frequency if it exists.
No
Does this word contain exactly 3 characters?
No
Skip
Yes
Is the last character of previous word same as the first character of current word and next word same as the last two characters of current word?
No
Yes
Is the previous word same as the first two characters of current word and the first character of next word same as the last character of current word?
Yes
Add previous word and next word to lexicon
No
Add current word to lexicon
Yes
Is any character not added to lexicon?
No
Yes
Find largest dts among the left characters of the sub-sentence
Is dts>=0?
No
Add left characters one by one to lexicon
Yes
Take this position as center, search bidirectionally, and bind characters together if the dts between two characters is greater than zero.
Keep searching until the dts is smaller than zero or no character left. Then, we get a candidate word.
Does this candidate word contain exactly two characters?
Yes
Add it to lexicon
No
Look up the dictionary and find all bigram-words within the candidate word
Does bigram-word exist?
No
Yes
Add all bigram-words and left single characters one by one to lexicon

Figure 2: Flow diagram of second iteration.

exactly two characters, add it to lexicon. Otherwise, look up the dictionary and find all bigram-words within the candidate word. If no bigram-word exists, add the candidate word to lexicon; otherwise, add all bigram-words to lexicon, and if some of characters of the candidate word are not added to lexicon, add left characters to lexicon one by one (take single character as a word). Overlap is allowed in this step. For example, $C_1C_2C_3C_4C_5$ is a candidate word. $C_1C_2$ and $C_2C_3$ are bigram-words contained in the dictionary. Eventually, $C_1C_2$, $C_2C_3$, $C_4$ and $C_5$ will be added to lexicon.

**Step 6:** Repeat step 5 until the largest dts is smaller than zero or no character left.

**Step 7:** If there are characters left, they are added one by one to lexicon. For instance, $C_1C_2C_3C_4C_5C_6C_7C_8C_9C_{10}$ is a sub-sentence and it gets processed from step 1 to step 6. $C_3$, $C_4$ and $C_{10}$ are left characters. Then, $C_3$, $C_4$ and $C_{10}$ are added to lexicon.

**Step 8:** Repeat step 1 to step 7 until all texts in corpus get processed.

To make the process clear, the corresponding flow diagram is given in Figure 2. In step 4, we skip over bigram-words in dictionary so as to avoid introducing more ambiguity. For a Chinese character string $C_1C_2C_3$, both $C_1C_2$ and $C_2C_3$ can be found in the dictionary. Under such a circumstance, we have less chance to make a right choice with the dictionary alone. For the predominance of bigram-words, words containing more than two characters, which have less ambiguity, can be added to lexicon directly within certain limitation. With statistical information, bigram-words can be extracted in step 5 to guarantee a certain accuracy. The reason we make such rules in step 5 will be further explained in Section 5.4. Eventually, we build a specific lexicon for CCWS from the *Twenty-Five Histories*.

### 4.2 Word segmentation

Now that there is a customized lexicon with frequency of each word, maximum probability algorithm is used for word segmentation. $w_i$ stands for a certain word and S stands for a sentence that contains n words. Ignore the relevance between words, and the probability of sentence S is defined as:

$$\mathrm{P(S)} = P(w_1) \times P(w_2) \times \cdots \times P(w_n)$$

Take the case of maximum value of P(S) as the optimal result of word segmentation. The probability of a certain word is defined as:

$$\mathrm{P}(w_i) = \frac{f(w_i)}{N}$$

$f(w_i)$ refers to the frequency of word $w_i$, and N is the total words of the corpus. Dynamic programming is applied for reducing the computation.

## 5. Experiments

We have done five comparison experiments to prove the effectiveness of our approach. Out of the fact that Classical Chinese literature is mostly composed of monosyllables, sentences segmented into individual characters are considered as the baseline of CCWS in our experiments. Before illustrating experiment results, the design of other three experiments will be described below.

### 5.1 Normalized Pointwise Mutual Information

Bouma (2009) introduced the normalized pointwise mutual information (NPMI) for collocation extraction. For character $C_1$ and $C_2$, the NPMI of them is defined as:

$$\mathrm{npmi}(C_1; C_2) = \frac{pmi(C_1; C_2)}{h(C_1; C_2)}$$

PMI of $C_1$ and $C_2$ is defined as:

$$\mathrm{pmi}(C_1; C_2) = \log \frac{p(C_1, C_2)}{p(C_1)p(C_2)}$$

$h(C_1;C_2)$ is the self-information, defined as:

$$\mathrm{h}(C_1; C_2) = -\log p(C_1, C_2)$$

NPMI ranges from -1 to 1, resulting in -1 for never occurring together, 0 for independence, and 1 for complete co-occurrence. In this experiment, if the NPMI between two characters is greater than zero, they are bound together. During the word extraction, do a scan over the sentence; characters bound together are added as a word to lexicon; characters left are added one by one to lexicon. The lexicon we get is named after 'lexicon1'.

### 5.2 Difference of t-score

The definition of dts is given in Section 4. In this experiment, if the dts between two characters is greater than zero, they are bound together. The process of word extraction is the same as that of previous experiment. The lexicon is named after 'lexicon2'.

### 5.3 Simple integration of dts with encyclopedia

In this experiment, the procedures are all the same as those of our approach described in Section 4, except the step 5. The candidate words are added straight to lexicon without the subsequent operations. The lexicon is named after 'lexicon3'. For 'lexicon3', we have counted words that contain more than two characters and are not included in the dictionary, as listed in Table 3 below.

| | 3-gram | 4-gram | 5-gram | 6-gram | 7-gram |
|---|---|---|---|---|---|
| Count | 28064 | 2215 | 35 | 1 | 0 |

Table 3: Count of N-gram

| 3-grams | 4-grams | 5-grams | 6-grams |
|---|---|---|---|
| 照英约<br>尝与刘<br>尚宫曰<br>少敏慧<br>尽其长<br>将安出<br>先帝崩<br>赤山湖<br>⋮<br>⋮ | 烧其船舰<br>尽其死力<br>物之失所<br>寻以母丧<br>辰星犯天<br>填星犯井<br>结为死党<br>虽悔何追<br>⋮<br>⋮ | 位于大明殿<br>衣画而裳绣<br>降者数万人<br>积度及分秒<br>为酷吏所陷<br>加上尊号曰<br>莫大于不孝<br>德厚者流光<br>⋮<br>⋮ | 周孝闵帝践祚 |

Table 4: Some samples of N-gram.

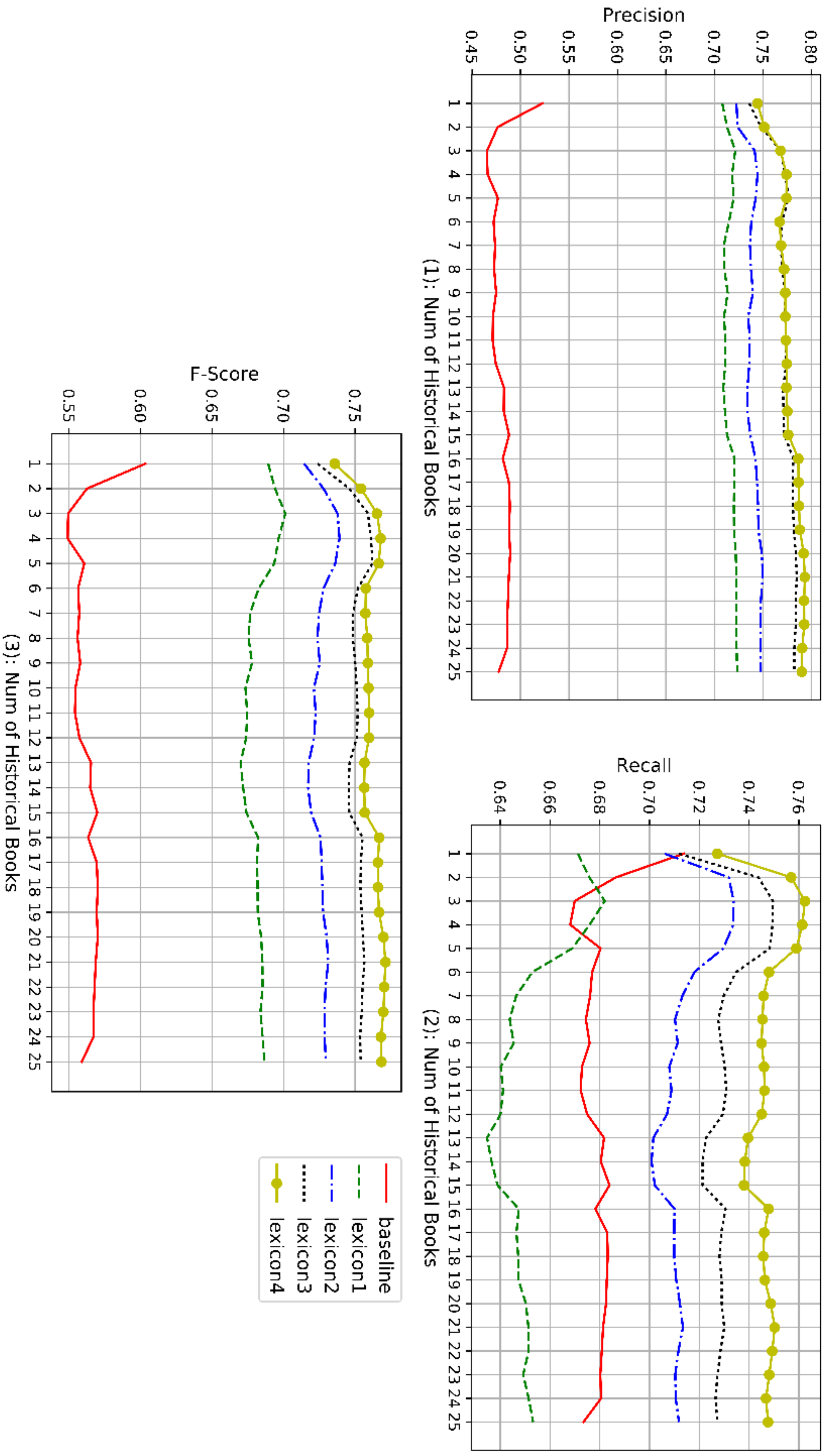

Figure 3: Precision, recall and F-Score of different approaches

Through our observation, a majority of these N-grams are frequent items instead of words, some of which are listed in Table 4. In order to enhance the validity of the lexicon, frequent items need to be processed ulteriorly. With further division of frequent items in our final solution, we get a lexicon named after ‘lexicon4’.

### 5.4 Experiment results

There are 20 commonly used function words (those have little lexical meaning and express grammatical relationships with other words within a sentence) in Classical Chinese literature, as listed in Table 5. The function words in the four lexicons are given the highest weight to ensure they can be correctly segmented out of the sentence in most occasions.

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 而 | 何 | 乎 | 乃 | 其 | 且 | 然 | 若 | 所 | 与 |
| 为 | 焉 | 也 | 以 | 矣 | 于 | 之 | 则 | 者 | 因 |

Table 5: Commonly used function words in ancient texts

As stated in Section 3.2, we randomly select sample texts, consisting of 32689 characters, from the *Twenty-Five Histories*. The size of samples picked out from each historical book is proportional to the size of the book for keeping the balance. In addition, the sample texts cover every aspect of the content. Figure 3 illustrates the precision, recall, as well as F-Score of five different approaches. The x-axis of the three subgraphs stands for the number of books used in experiments. For instance, “4” means we used the first 4 historical books that were written in chronological order (*Records of the Grand Historian*, *Book of Han*, *Book of Later Han*, *Records of the Three Kingdoms*) to measure the performance. As for the reason that the performance is showed in accumulation, it can prove the stability and feasibility of our approach over literature of different eras (There are unique words and expressions in each dynasty, some of which only appear within a certain period of time). In other words, it’s similar to one-size-fits-all approach, with which we don’t have to create lexicon for each book.

Judging from the F-Score subgraph, it’s obvious that our approach surpasses the others, and it can keep a high stable F-Score with the increment of test data. Over the whole test data, the F-Score of our approach is 76.84%. Recall subgraph shows that the drawback of NPMI results in the drop of the recall rate of ‘lexicon1’. Without an explicit threshold, it’s not possible to extract high quality multisyllabic words with NPMI. Besides, the conspicuous drop of recall rate of baseline method verifies the phenomenon that individual character is mostly used to represent a monosyllabic word in Archaic Chinese and the number of multi-syllables increases gradually as time goes on.

An example of CCWS over a specific sentence is listed in Table 6. Approach with ‘lexicon3’ or ‘lexicon4’ is able to identify the word ‘勃然作色’ correctly for the combination with Baidu Baike. The word is a Chinese idiom of which the definition is included in Baidu Baike. However, with the statistical information alone, this 4-gram word is not extracted out of the corpus. As for the trigrams ‘太息曰’ and ‘虽不肖’, which are not included in Baidu Baike, only method with ‘lexicon4’ correctly segments them into two words respectively. If the trigram is not in Baidu Baike while a bigram, being part of it, is included, we tend to believe that the trigram is composed of a bigram-word as well as an individual character (correct segmentation is ‘太息|曰’ and ‘虽|不肖’). For low precision of bigram-word extraction, our method wrongly identifies bigram-words ‘事秦’ and ‘今主’ in the instance. In step 5 of our approach, we add overlapped distinct bigram-words to the lexicon, which results in the deficiency in precise extraction of bigram-word. We have attempted to pick only one out of two overlapped bigram-words according to their dts, but the final result is not improved. Increasing the accuracy of bigram-word extraction is our future work.

<table>
<tr><td>Original</td><td>于是韩王勃然作色，攘臂瞋目，按剑仰天太息曰：“寡人虽不肖，必不能事秦。今主君诏以赵王之教，敬奉社稷以从。”</td></tr>
<tr><td>Standard</td><td>于是|韩王|勃然作色|，|攘臂瞋目|，|按剑|仰天|太息|曰|：|“|寡人|虽|不肖|，|必|不能|事|秦|。|今|主君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|”</td></tr>
<tr><td>Lexicon1</td><td>于|是|韩|王勃|然|作色|，|攘|臂|瞋目|，|按剑|仰天|太息曰|：| “|寡人|虽不肖|，|必不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|”</td></tr>
<tr><td>Lexicon2</td><td>于是|韩王|勃|然|作色|，|攘臂|瞋目|，|按剑|仰天|太息曰|：| “|寡人|虽不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|”</td></tr>
<tr><td>Lexicon3</td><td>于是|韩王|勃然作色|，|攘臂|瞋目|，|按剑|仰天|太息曰|：| “|寡人|虽不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|”</td></tr>
<tr><td>Lexicon4</td><td>于是|韩王|勃然作色|，|攘臂|瞋目|，|按剑|仰天|太息|曰|：| “|寡人|虽|不肖|，|必|不能|事秦|。|今主|君|诏|以|赵王|之|教|，|敬奉|社稷|以|从|。|”</td></tr>
</table>

Table 6: CCWS example

## 6. Conclusion

In this paper we proposed a pragmatic approach combining the difference of t-score and encyclopedia (Baidu Baike) for CCWS over literature of different eras. The F-Score over the whole evaluation data set is 76.84%. To a certain degree, this approach releases researchers from labor intensive work, like constructing corpus, and makes it possible to build standard large-scale shareable marked-up corpora for the study of Classical Chinese literature. It also facilitates the research of Classical Chinese literature throughout the whole history instead of over a specific period of time or particular books. Besides, the scale of Baidu Baike we used in experiments is just two-thirds of the latest version, which means the performance of our approach can be better. With the Baidu Baike gradually being complete in terms of knowledge about Classical Chinese literature, the performance will be highly improved. This is another merit of our approach. After all, new content won’t be generated for the obsoleteness of Classical Chinese.

## 7. Acknowledgements

This work is supported by National Natural Science Foundation of China (No. 61070066, No. 61379073), and the CADAL (China Academic Digital Associative Library) Project and Research Center, Zhejiang University.

## 8. Bibliographical References

Bouma, G. (2009). Normalized (Pointwise) Mutual Information in Collocation Extraction. *Proceedings of GSCL*, pp. 31--40.

Cai, D. and Zhao, H. (2016). Neural Word Segmentation Learning for Chinese. *Association for Computational Linguistics*, pp. 409--420.

Chang, J.S. and Su. K.Y. (1997). An Unsupervised Iterative Method for Chinese New Lexicon Extraction. *International Journal of Computational Linguistics & Chinese Language Processing*, 2(2):97-147.

Chen, K.J. and Ma, W.Y. (2002). Unknown Word Extraction for Chinese Documents. *COLING '02 Proceedings of the 19th international conference on Computational linguistics*, (1):1-7.

Chen, X.C., Qiu, X.P., Zhu, C.X. and Huang, X.J. (2015a). Gated Recursive Neural Network for Chinese Word Segmentation. *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing*, pp. 1744--1753.

Chen, X.C., Qiu, X.P., Zhu, C.X., Liu, P.F. and Huang, X.J. (2015b). Long Short-Term Memory Neural Networks for Chinese Word Segmentation. *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pp. 1197--1206.

Church, K.W., Gale, W.A., Hanks, P. and Hindle, D. (1991). Using Statistics in Lexical Analysis. *Lexical Acquisition: Exploiting Online Resources to Build a Lexicon, Publisher: Lawrence Erlbaum, Editors: Uri Zernik*, pp. 115--164.

Duan, L., Han, F. and Song, J.H. (2012). A Comparative Study on the Automatic Extraction of Two-character Word from Ancient Chinese. *Journal of Chinese Information Processing*, 26(04):34-42.

Fang, A.C., Lo, F. and Chinn, C.K. (2009). Adapting NLP and corpus analysis techniques to structured imagery analysis in classical Chinese poetry. *The Workshop on Adaptation of Language Resources and Technology To New Domains Association for Computational Linguistics*，pp. 27--34.

Feng, H.D., Chen, K., Deng, X.T. and Zheng, W.M. (2004). Accessor variety criteria for Chinese word extraction. *Computational Linguistics*, 30(1):75-93.

Lee, J. and Wong, T.S. (2013). Glimpses of Ancient China from Classical Chinese Poems. *COLING 2012: Posters*, pp. 621--632.

Li, B., Xi, N., Feng, M. and Chen, X. (2012). Corpus-Based Statistics of Pre-Qin Chinese. *Chinese Conference on Chinese Lexical Semantics Springer-Verlag*, (7717):145-153

Liu, C.L., Wang, H., Cheng, W.H., Hsu, C.T. and Chiu, W.Y. (2015). Color Aesthetics and Social Networks in Complete Tang Poems: Explorations and Discoveries. *Computer Science*.

Luo, S.F. and Sun, M.S. (2003). Chinese Word Extraction Based on the Internal Associative Strength of Character Strings. *Journal of Chinese Information Processing*, 17(03):10-15.

Ma, W.Y. and Chen, K.J. (2003). A bottom-up merging algorithm for Chinese unknown word extraction. *SIGHAN '03 Proceedings of the second SIGHAN workshop on Chinese language processing*, (17):31-38.

Mei, L.L., Huang, H.Y., Wei, X.C., Yuan, P. and Mao, X.L. (2015). FCL: A New Network Words Extraction Approach Based on Statistical Language Knowledge. *Social Media Processing, Springer Singapore*.

Ouyang, J. (2016). Visual Analysis and Exploration of Ancient Texts for Digital Humanities Research. *Journal of Library Science in China*, (02):66-80.

Pei, W.Z., Ge, T. and Chang B.B. (2014). Max-Margin Tensor Neural Network for Chinese Word Segmentation. *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics*, pp. 293--303.

Peng, F.C., Feng, F.F. and McCallum, A. (2004). Chinese segmentation and new word detection using conditional random fields. *In Proceedings of COLING*, pp. 562--568.

Qiu, B., Huang, P.J. (2008). Study on the Trend of Ancient Chinese Words Based on the Word Automatic Segmentation. *Microcomputer Information*, 24(24):100-102.

Shen, M., Kawahara, D. and Kurohashi, S. (2016). Chinese Word Segmentation and Unknown Word Extraction by Mining Maximized Substring. *Journal of Natural Language Processing*, 23(3):235-266.

Shi, M., Li, B. and Chen, X.H. (2010). CRF Based Research on a Unified Approach to Word Segmentation and POS Tagging for Pre-Qin Chinese. *Journal of Chinese Information Processing*，24(2):39-45.

Sproat, R. and Shih, C. (1990). A Statistical Method for Finding Word Boundaries in Chinese Text. *Computer Processing of Chinese & Oriental Languages*, 4(4):336-351.

Sun, M.S., Shen, D.Y. and Benjamin, K.T. (1998). Chinese Word Segmentation without Using Lexicon and Hand-crafted Training Data. *Meeting of the Association for Computational Linguistics and, International Conference on Computational Linguistics Association for Computational Linguistics*, 48(2):1265-1271.

Tang, L.X., Geva, S., X, Y. and Trotman, A. (2009). Word Segmentation for Chinese Wikipedia Using N-Gram Mutual Information. *Plos Medicine*, 2(7):576-582.

Tseng, H.H., Chang, P.C, Andrew, G., Jurafsky, D. and Manning, C. (2005). A conditional random field word segmenter for sighan bakeoff 2005. *In proceedings of the fourth SIGHAN workshop*, pp. 168--171.

Xu, Y.M. (2016). Some Visualization Approaches to the Study of Classical Chinese Literature: A Case Study on Tang Xianzu. *Journal of Zhejiang University (Humanities and Social Sciences Online Edition).*

Xue, N.W. (2003). Chinese word segmentation as character tagging. *International Journal of Computational Linguistics & Chinese Language Processing*, 8(1):29-47.

Zhang, C.X., Niu, Z.D., Jiang, P. and Fu, H.P. (2012). Domain-specific term extraction from free texts. *International Conference on Fuzzy Systems and Knowledge Discovery IEEE*, pp. 1290--1293.

Zhang, H.J., Huang, H.Y., Zhu, C.Y. and Shi, S.M. (2010). A Pragmatic Model for New Chinese Word Extraction. *International Conference on Natural Language Processing and Knowledge Engineering IEEE*, pp. 1--8.

Zhang, W., Yoshida, T., Tang, X.J and Ho, T.B. (2009). Improving effectiveness of mutual information for substantival multiword expression extraction. *Expert Systems with Applications*, 36(8):10919-10930.

Zhang, Y. and Clark, S. (2007). Chinese Segmentation with a Word-Based Perceptron Algorithm. *Proceedings of the 45th Annual Meeting of the Association for Computational Linguistics*.

Zheng, X.Q., Chen, H.Y. and Xu, T.Y. (2013). Deep learning for Chinese word segmentation and POS tagging. *Conference on Empirical Methods in Natural Language Processing*, pp. 647--657.