CFS-based Unigram Language Model, a Brave New Approach to Traditional N-gram Language Models

以中文常用字串為基礎的 unigram 語言模型，一個優於傳統語言模型的新方法

Yih-Jeng Lin(林義証)
Department of Information Management
Chien Kuo Institute of Technology
Changhua, 500 Taiwan
yclin@ckit.edu.tw

Ming-Shing Yu(余明興)
Department of Applied Mathematics
National Chung-Hsing University
Taichung, 40227 Taiwan
msyu@dragon.nchu.edu.tw

Abstract

This paper introduces a new concept, the Chinese frequent strings (CFS) based unigram language model, which is in many respects superior to the traditional language model (LM). Important properties of CFSs and applications in Chinese natural language processing (NLP) will be revealed in this paper. We have proposed a methodology for extracting Chinese frequent strings, which contain unknown words, from a Chinese corpus. We found that CFSs contain many 4-gram characters, 3-gram words, and higher n-grams. Such information can only be derived with an extremely large corpus in a traditional language model. In contrast to using a traditional LM, we can achieve high precision and efficiency by using CFSs to solve Chinese toneless phoneme-to-character conversion and to correct Chinese spelling errors with a small training corpus. An accuracy of 92.86% was achieved for Chinese toneless phoneme-to-character conversion. An accuracy of 87.32% was achieved for Chinese spelling error correction. We used a traditional lexicon, namely the ASCED (Academia Sinica Chinese Electronic Dictionary) provided by Academia Sinica, Taiwan, and the word bigram language model to solve the two abovementioned problems. We achieved accuracies of 66.9% and 80.95% respectively for Chinese toneless phoneme-to-character conversion and Chinese spelling error correction.

Keywords: *Chinese frequent strings, Chinese toneless phoneme-to-character, Chinese spelling*

1. Introduction

There are an increasing number of new or unknown words used on the Internet. Such new or unknown words are called "out of vocabularies" (OOV) and they are not listed in traditional dictionaries. Many researchers overcome the problems which are caused by OOV by using N-gram LMs. N-gram LMs have many useful applications in NLP (Yang, 1998). In Chinese NLP tasks, the word bigram LM is used by many researchers. To get predictable probabilities in training, a corpus size of about 8000^2 (8000 is the approximate number of words of ASCED) = $6.4*10^9$ words is required. It is not easy to find such a corpus at the present time.

A small-size corpus will lead too many unseen events when using N-gram LMs. Although we can apply some smoothing strategies, such as Witten-Bell interpolation or Good-turning method (Wu and Zheng, 2001) to estimate the probabilities of unseen events, it will be of no use when the size of training corpus is limited. In our observations, many unseen events of N-gram LMs are unknown words or phrases. Such unknown words and phrases cannot be found in the dictionary. For example, the term "**小企鵝**" (a little penguin) is a word bigram pattern which consists of two words "**小**" (little) and "**企鵝**" (penguin). Many researchers show that using phrases is a good way to enhance the performance of LMs (Jelinek, 1990; Suhm and Waibel, 1994). Another example is the term "**週休二日**" (two days off per week). Such an expression is presently popular in Taiwan. We cannot find this term in a traditional dictionary. The term "**週休二日**" is a 4-gram word pattern which consists of four words "**週**" (a week), "**休**" (to rest), "**二**" (two), and "**日**" (day). A 4-gram word LM and a large training corpus are required to record the data of such terms. Such a 4gram word LM has not been applied to Chinese NLP practice and such a huge training corpus cannot be found at present. Alternatively, we can record the specifics of the term "**週休二日**" by using a CFS-based unigram LM with relatively small training data which contains the specified term twice or more. Such training data could be recorded in one or two news articles containing hundreds of Chinese characters.

Each CFS can contain the information of n-gram on the word level, where 'n' can be up to

3. It must be noted that most combinations cannot be found in a word bigram language model. Such unseen events may degrade the performance of many NLP tasks. When a word bigram appears twice or more in a language model, it is likely that this bigram will also be a CFS, especially when its count is high. In our study, we will show that using the CFS-based unigram model can achieve better results than using the traditional word bigram model when training a small-size corpus.

The organization of this paper is as follows. Section 2 gives some properties and distributions of CFSs. We make a comparison between CFS and an n-gram LM (language model). Section 3 shows that using CFSs with a unigram LM can achieve higher accuracy than the use of a traditional lexicon with word bigram LMs in two challenging examples of Chinese NLP. Finally, Section 4 presents the discussion and conclusion.

2. The Properties of CFS versus LM and ASCED

There is a training corpus of 59 MB (about 29.5M Chinese characters) contained in this paper. The training corpus contains a portion of ASBC and many daily news for Internet. In this section, we will present the properties of CFSs. Compared with language models and ASCED, CFSs have some important features. We will describe 439,666 CFSs in subsection 2.1.

2.1 Extracting CFSs

We extracted CFSs from a training corpus, the content size of which was 29.5M characters. The training corpus also included a portion of the Academia Sinica Balanced Corpus (Chen et al., 1996). The method of extracting CFSs is as follows. First, we will offer some notations.

C: The training corpus,

M : The MayBe database, each item in *M* may be a CFS.

S: Set of CFS,

w, q, l : String patterns,

L_w : The number of characters in the string pattern w ,

F_w : The frequency of w which occurs in C,

N_w : The net frequency of w which occurs in C.

Then we know that

$$\forall w \in \mathrm{C} \text{ with } L_w \geq 1 \text{ and } L_w \leq 12,\ w \in M \text{ iff } F_w \geq 2$$

and

$$\forall w \in M, \text{if } N_w \geq 2 \text{ then } w \in \mathrm{S}.$$

First, we will define a notation $b_w (\subseteq M)$ and an operation $q = w \oplus l$. Next, we will show a formula for N_w . Consider $w = (w_1 w_2 w_3 ... w_{L_w})$, where w_i is the *i*th character in the string w . We define b_w to be the strings $q = (q_1 q_2 q_3 .. q_{L_w} q_{L_w+1})$ in M such that w is a substring of q , i.e., $w = (q_1 q_2 q_3 .. q_{L_w})$ or $w = (q_2 q_3 .. q_{L_w} q_{L_w+1})$.

The operation $q = w \oplus l$ is defined as follows. When the last $L_w - 1$ characters of w are the same as the first $L_l - 1$ characters of l , q is the concatenation of w and the last character of l . For instance, *q*="**中興大學**" if w =" **中興大**" and l =" **興大學**". If the above condition does not hold, i.e., $(w_2 w_3 ... w_{L_w}) \neq (l_1 l_2 ... l_{L_l - 1})$ then q is an empty string. The net frequency of w is defined as：

$$N_w = F_w - \sum_{q \in b_w} F_q + \sum_{\forall q, l \in b_w}^{q \neq l} F_{q \oplus l},$$

where

$$F_{q \oplus l} = \begin{cases} F_{q \oplus l}, & \text{if } q \oplus l \in M \\ 1, & \text{if } q \oplus l \notin M \text{ and } q \oplus l \in C \\ 0, & \text{if } q \oplus l \notin M \text{ and } q \oplus l \notin C \end{cases}.$$

Consider the following Chinese text, we can extract some CFSs from the text. Such as "**國有眷舍**", "**國立大學**", "**處理**", and so on.

"**這份聲明指出，目前高等教育資源嚴重不足，政府成立「國家資產經營管理委員會」，研訂收回大學國有眷舍土地的政策，是短視近利的作法。此外，國立大學也擔心這些取得不易的土地，如果落入財團手裡，將來擴充校地可能要花更多錢，而且可能破壞學術社群。因此，國立大學要求依「公教分離」原則，將國立大學眷舍土地的處理排除在國有眷舍處理的範圍之外。**"

The distribution of length of the CFSs is shown in the second column of Table 1. The total number of CFSs we extracted is 439,666. In contrast to the second column of Table 1, we show the distribution of the length of the words in the ASCED in the forth column of Table 1. We found that the number of three-character

CFSs in our CFS lexicon is the greatest, while the number of two-character words in ASCED is the greatest. Many meaningful strings and unknown words are collected in our CFSs. Such a CFS usually contains more than two characters. Some examples are "**小企鵝**" (a little penguin), "**西醫師**" (modern medicine), "**佛教思想**" (the thought of Buddhism), "**樂透彩券**" (lottery), and so on. The above examples cannot be found in the ASCED, yet they frequently appear in our training corpus.

2.2 CFSs vs. LMs

Since CFSs are frequent strings used by people, a CFS, such as "**大學教授**" (professors of a university), may contain more characters than a word defined in ASCED. A CFS may contain two or more words. If a CFS contains two words, we say that this CFS is a word bigram CFS. If a CFS contains three words, we say that this CFS is a tri-gram word CFS. Figure 1 is the distributions of CFSs according to word n-grams. The words are defined in the ASCED.

From Figure 1, it can be shown that a CFS may contain more than 3 words. Many researchers in Chinese NLP used bigram word LMs (Yang, 1998) as a basic LM to solve problems. A very large corpus is required to train a 3-gram word LM, while our CFS-based unigram model does not need such a large corpus. We also found that a CFS contains 2.8 words on average. This shows that a CFS contains more information than a bigram word LM. In our experiment, we also found that the average number of characters of a word bigram is 2.75 and the average number of characters of a CFS is 4.07. This also shows that a CFS contains more information than a word bigram.

2.3 CFSs vs. ASCED

In this subsection, we will make a comparison between our CFSs and the ASCED. Table 1 and Figure 2 are the distributions of length of our CFSs and the ASCED. Comparing the distribution of lengths of CFSs and the ASCED, we found that the average number of characters of a word in the ASCED is 2.36, while the average number of characters in a CFS is 4.07. Examining Figure 2, we noticed that most of the words in the ASCED are 2-character words, while the largest portion of CFSs are 2-character CFSs, 3-character CFSs, 4-character CFSs, and 5-character CFSs. This shows that our CFSs contain many 4-gram and 5-gram characters. To train 4-gram and 5-gram character LMs requires a large training corpus. We also found that the number of one-character CFSs is fewer than that of the ASCED. This shows that using the CFSs can eliminate some ambiguities in Chinese PTC and Chinese CTP.

There are 31,275 CFSs which are words in the ASCED. We compared the distribution of the length of these 31,275 CFSs with the distribution of the ASCED. A comparison is shown in Figure 3. Note that the distribution of the ASCED is listed in the fifth column of Table 1. We found that the distribution of these 31,275 CFSs is similar to the distribution of the ASCED. We conjectured that if the corpus is large enough, we can find most of the words in the ASCED.

2.4 Comparing the normalized perplexity

Normalized perplexity (Yang, 1998) or perplexity (Rabiner and Juang, 1993) is an important and commonly used measurement in language models.

We use a testing corpus to compute the normalized perplexities within the CFS-based unigram LM and the word bigram LM. The size of the testing corpus was 2.5M characters, and the testing corpus does not contain the training corpus mentioned in subsection 2.1. We used the same training corpus mentioned in subsection 2.1 to extract CFSs and to train the word bigram LMs. Each word in the word bigram LM is defined in the ASCED. We use the Good-Turning smoothing method to estimate the unseen bigram events. The normalized perplexity is 78.6 by using the word bigram LM. The normalized perplexity becomes 32.5 by using the CFS-based unigram LM. This shows that the CFS-based unigram LM has a lower normalized perplexity. That is to say, using the CFS-based unigram LM is better than a traditional word bigram LM, especially with a small-size training corpus.

3. Applications of the CFS-based Unigram LM in Two Difficult Problems

In a previous study (Lin and Yu, 2001), we showed that using CFSs and the ASCED as the dictionary with the unigram language model can achieve good results in two applications of Chinese NLP. These two applications are Chinese character-to-phoneme (CTP) conversion and Chinese phoneme-to-character (PTC) conversion. The accuracies were 99.7% for CTP conversion and 96.4% for PTC conversion. The size of the training corpus in our previous research is 0.5M characters. There were 55,518 CFSs extracted from the training corpus. In this paper, we will solve two challenging problems of Chinese NLP with a larger training corpus. The two issues are Chinese toneless

phoneme-to-character (TPTC) conversion and Chinese spelling error correction (SEC).

3.1 Chinese toneless phoneme-to-character conversion

The first task is Chinese TPTC conversion. The lexicon we used is comprised of the 439,666 CFSs mentioned in Section 2.1. This task is more complex than traditional Chinese phoneme-to-character conversion. There are five tones in Mandarin. They are high-level (1st tone), high-rising (2nd tone), low-dipping (3rd tone), high-falling (4th tone), and neutral tone (National Taiwan Normal University, 1982; Lin and Yu, 1998). There are a total of 1,244 possible syllables (combinations of phonetic symbols), and there are a total of 408 possible toneless syllables (Hwang and Chen, 1994). Therefore, each toneless syllable has about 1,244/408=3.05 times the number of characters of a tonal syllable. The average length of a sentence in our training corpus is 8 characters per sentence. The number of possibilities in Chinese TPTC conversion is about 3.05^8=7489 times that of Chinese PTC conversion. We also found that on average a tonal syllable contains about 21.40 characters and a toneless syllable contains about 62.6 characters in the training corpus. This shows that Chinese TPTC conversion is more difficult than Chinese PTC conversion.

For example, consider the sequence of toneless phonemes "ㄧ ㄕ /yi shi/", there are many words whose toneless phonemes are the same as "ㄧ ㄕ". The words are "**議事**(ㄧˋㄕˋ /yi4 shi4/)" (discuss official business), "**意識**(ㄧˋㄕˋ/yi4 shi4/)" (consciousness), "**醫師**(ㄧ ㄕ /yi1 shi1/)" (doctor), "**儀式** (ㄧˊㄕˋ /yi2 shi4/)" (ceremony), "**一時** (ㄧˋㄕˊ /yi4 shi2/)" (for a short while), "**衣飾** (ㄧ ㄕˋ /yi1 shi4/)" (clothing), and so on. It is reasonable to do TPTC conversion by the traditional n-gram word LM. We also can accomplish the above task by using longer CFSs with the unigram model. For example, we can use related useful CFSs like "**干擾議事**" and "**公開儀式**" to decide what the sequence of toneless phonemes "ㄧ ㄕ /yi shi/" means. Note that we have collected such useful CFSs, which may contain two or more words defined in a traditional lexicon, from the training data.

We use the 439,666 CFSs which are extracted from the training data with a size of 29.5M characters as the system dictionary. The size of the outside testing data is 2.5M characters. In our TPTC module, we initially searched the system dictionary to assess all the possible CFSs according to the input of toneless phonemes. Such possible CFSs constitute a CFS lattice. We applied a dynamic programming methodology to find the best path in the CFS lattice. The best path is the sequence of CFS-based unigrams with the highest probability.

The precision rate is 92.86%. The precision rate is obtained by the formula (total number of correct characters) / (total number of characters). The processing time is 12 ms/character. We also applied the dictionary in our previous research (Lin and Yu, 2001) to test the data which was 2.5M characters in size. The dictionary is the combination of the ASCDE and the 55,518 CFSs. The precision rate is 87.3% in solving the Chinese TPTC problem. This indicates that if we can collect more CFSs, we can obtain higher accuracy.

In this task, we also applied the bigram word LM with the ASCED. The size of the training corpus is the same as the corpus mentioned in Section 2.1. Note that the size of the corpus is 29.5M characters. The Good-Turning smoothing method is applied here to estimate the unseen events. The precision rate is 66.9% and the processing time is 510 ms/character. We propose that by using CFSs with the unigram LM, the precision rate is much higher (92.8 % vs. 66.9%) and the processing time is far less (12 ms/character vs. 510 ms/character) than the traditional bigram word LM.

3.2 The Chinese spelling error correction issue

We applied the 439,666 CFSs to the Chinese SEC issue (Chang, 1994). Chinese SEC is a challenging undertaking in Chinese natural language processing tasks. A Chinese SEC system should correct character errors for the input sentence. To prevent ambiguity, we limit our Chinese SEC problem to the following two hypotheses: (1) the sentences are input using the Cang-Jie Chinese input method, (2) there is no more than one character error in an input sentence.

The reasons why we propose the above two hypotheses are (1) our Chinese SEC system is designed for practiced typists, (2) the Cang-Jie Chinese input method is a popular method which is widely used in Taiwan, (3) there is likely only one character error in a sentence for a practiced typist, and (4) we can easily apply the methodology of this research to other Chinese input or processing systems. Our methodology for the Chinese SEC is shown in the SEC Algorithm

Algorithm 1.

Input: A sentence *S* with no more than one incorrect character.

Output: The corrected sentence for the input

sentence *S*.
Algorithm:

Step 1: For each *ith* character in *S*, find the characters whose Cang-Jie codes are similar to the code of the *ith* character. Let *C* be the set consisting of such characters. *C* is called the 'confusing set'.

Step 2: Replace each character in *C* for the *ith* character in *S*. There will be a 'maybe' sentence S_1. Find the probability of S_1 by CFSs with the unigram LM. Record the maybe sentence with the highest probability.

Step 3: For each character in *S*, repeat Step 1 and Step 2.

Step 4: Output the 'maybe' sentence with the highest probability found in Steps 1, 2, and 3.

The characters with similar Cang-Jie codes define the confusing set in Algorithm 1. We constructed the confusing set for each Chinese character by the five rules listed in Table 2. The longest common subsequence (LCS) algorithm is a famous algorithm which can be found in most computer algorithm books like (Cormen et al., 1998).

The unigram language model determined the probability of each sentence. We used the 439,666 CFSs as our dictionary. There are 485,272 sentences for the outside test. No more than one character in each sentence is replaced by a similar character. Both the location of the replaced character and the similar character are randomly selected. The precision rate was 87.32% with the top first choice. The precision rate was defined as (the number of correct sentences) / (the number of tested sentences). The top 5 precision rates are listed in Table 3. The precision rate of the fifth choice is about 95% in Table 3. This shows that we can offer five possible corrected sentences for users in practice. The precision rate is 97.03% in determining the location of the replaced character with top first choice.

Table 4 shows examples where the second choice is the correct answer. We found that each first choice in Table 4 is reasonable, too. Note that the probabilities of the first choice are slightly higher than the second choice.

We also applied the ASCDE with bigram word LMs in computing the probability for each possible sentence. The size of the training corpus was 29.5M characters which is the same as the training corpus mentioned in Section 2.1. We also used the Good-Turning smoothing method to estimate the unseen bigram events. The precision rate is shown in Table 5. The precision rate is 80.95% with top first choice.

From Table 3 and Table 5, we can find that using CFSs with a unigram LM is better than using the ASCED with a bigram word LM. The advantage is the high precision rate (87.32% vs. 80.95%) and the low processing time (55 ms/character vs. 820 ms/character).

6. Discussion and Conclusion

In this paper, we found that the CFS-based unigram LM is superior to traditional N-gram LMs. While the size of a corpus using the CFS-based unigram LM can be far smaller than that needed in traditional N-gram LMs, the applications show that the results are better by using the CFS-based unigram LM than by using an n-gram LM. We showed some important properties of Chinese frequent strings. We also used these properties in applications. The properties and applications are listed as follows:

(1) The distribution of CFSs which can be found in the ASCED is similar to the distribution of the ASCED. This shows that we uniformly extracted a portion of the ASCED from the training corpus as CFSs. We predicted that if we could extract more and more CFSs, some of these CFSs may be words from the ASCED. CFSs contain the distribution information from the ASCED.

(2) Among the distribution of length of CFSs, the portions of 2-character, 3-character, 4-character, and 5-character CFSs are more than 10% of the sample. Also, the average length of CFSs is 4.07 characters. If we want to train a 4-gram character LM, it requires a corpus size of about 5000^4 (5000 is the approximate number of frequently used Chinese characters) = $6.25*10^{14}$ characters. At present, we cannot find such a corpus.

(3) Compared to an n-gram word LM, the portions of 2-gram, 3-gram, and 4-gram CFSs are more than 10%. In addition to this, the average number of words in a CFS is 2.75. If we want to train a 3-gram word LM, a corpus of size of about 8000^3 is required (8000 is the approximate number of words of ASCED) = $5.12*10^{14}$ words. At present, we cannot find such a

corpus.

(4) We can conclude that CFSs contain important information from the ASCED and LM by the three characteristics mentioned above. We obtained such information without using a very large corpus. We can achieve higher accuracy by using a CFS-based unigram LM with a small corpus than by using a traditional n-gram LM with smoothing methods.

(5) We achieved high precision rates in both Chinese TPTC and Chinese SEC problems by using a CFS-based unigram LM. The processing is also more efficient than using a bigram LM. We think that CFS-based unigram LMs have applications in many other Chinese NLP scenarios.

Acknowledgements

We would like to acknowledge Academia Sinica for its ASBC corpus, ASCED dictionary, and Sinica Treebank. We also extend our gratitude to the many news companies for distributing their files on the Internet.

References

1. C. H. Chang, "A Pilot Study on Automatic Chinese Spelling Error Correction," *Communication of COLIPS*, Vol. 4, No. 2, 1994, pp. 143-149.
2. K. J. Chen, C. R. Huang, L. P. Chang, and H. L. Hsu, "SINICA CORPUS: Design Methodology for Balanced Corpora," *Proceeding of PACLIC 11th Conference*, 1996, pp. 167-176.
3. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, "Introductions to Algorithms," The MIT Press, 1998.
4. S. H. Hwang and S. H. Chen, "A Neural Network Based F0 Synthesizer for Mandarin Text-to-Speech System," IEE Porc. Vis. Image Signal Process, Vol. 141, No. 6, Dec., 1994, pp. 384-390.
5. F. Jelinek, "Self-organized Language Modeling for Speech Recognition," Readings in Speech Recognition, pp. 450-506, Ed. A Wabel and K. F. Lee. Morgan Kaufmann Publishers Inc., San Mateo, California, 1990.
6. Y. J. Lin and M. S. Yu, "An Efficient Mandarin Text-to-Speech System on Time Domain," *IEICE Transactions on Information and Systems*, Vol. E81-D, No. 6, 1998, pp. 545-555.
7. Y. J. Lin and M. S. Yu, "Extracting Chinese Frequent Strings Without a Dictionary From a Chinese Corpus And its Applications," *Journal of Information Science and Engineering*, Vol. 17, No. 5, 2001, pp. 805-824.
8. National Taiwan Normal University, "Mandarin Phonetics," National Taiwan Normal University Press, Taipei, Taiwan, 1982.
9. L. R. Rabiner and B. H. Juang, "Fundamentals of Speech Recognition," Prentice Hall Co. Ltd., 1993.
10. B. Suhm and A. Waibel, "Toward Better Language Models for Spontaneous Speech," Proc. ICSLP, 1994, pp. 831-834.
11. Jian Wu and Fang Zheng, "On Enhancing Katz-Smoothing Based Back-Off Language Model," *International Conference on Spoken Language Processing,* 2001, pp. I-198-201.
12. K. C. Yang, "Further Studies for Practical Chinese Language Modeling," Master Thesis, Department of Electrical Engineering, National Taiwan University, Taipei, Taiwan, June 1998.

Table 1. The distributions of length of CFSs and ASCED

Number of characters in a CFS or a word	Number of CFSs of that length in our CFS dictionary	Percentage	Number of words of that length in ASCED	Percentage
1	3,877	0.88%	7,745	9.57%
2	69,358	15.78%	49,908	61.67%
3	114,458	26.03%	11,663	14.41%
4	113,005	25.70%	10,518	13.00%
5	60,475	13.75%	587	0.73%
6	37,044	8.43%	292	0.36%
7	19,287	4.39%	135	0.17%
8	11,494	2.61%	66	0.08%
9	6,588	1.50%	3	0.004%
10	4,080	0.93%	8	0.006%

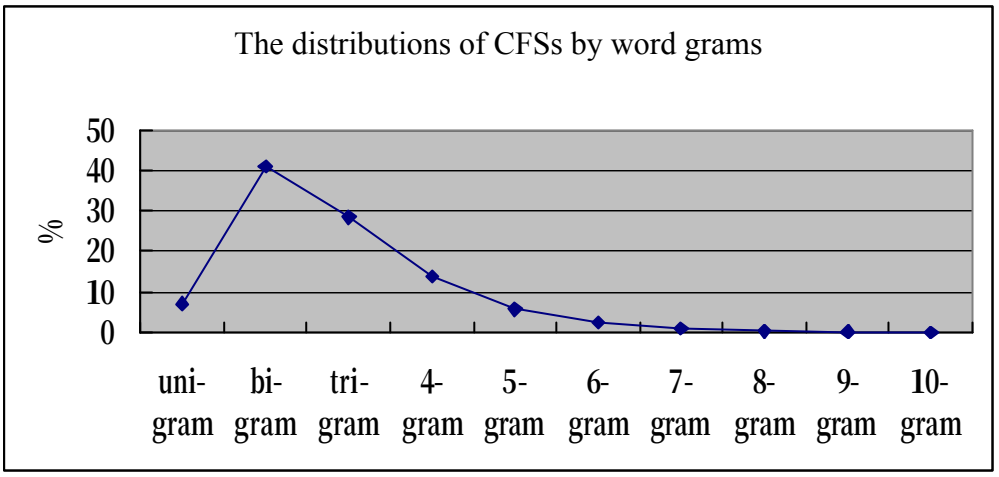

Figure 1. The distributions of CFSs by word grams

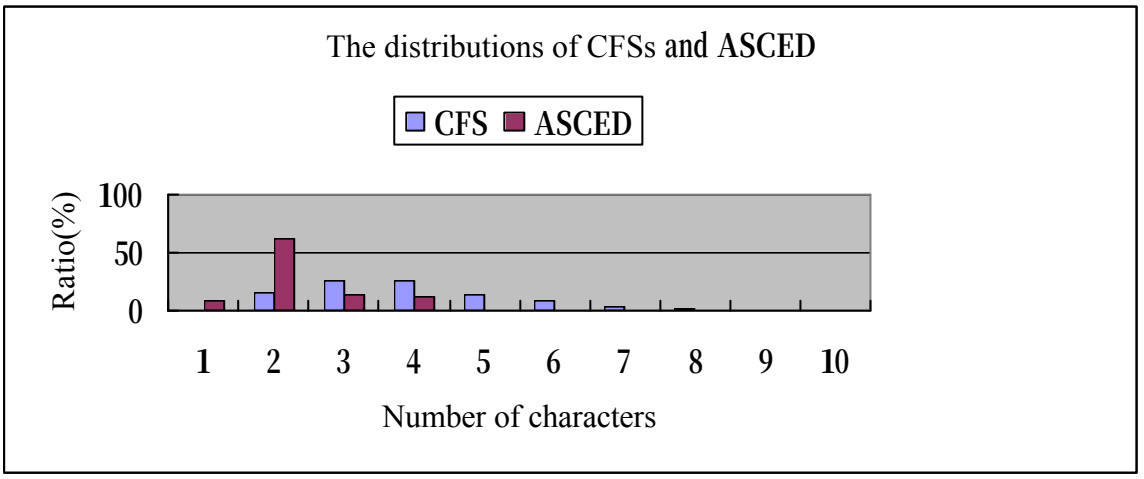

Figure 2. The distributions of length of CFSs and ASCED

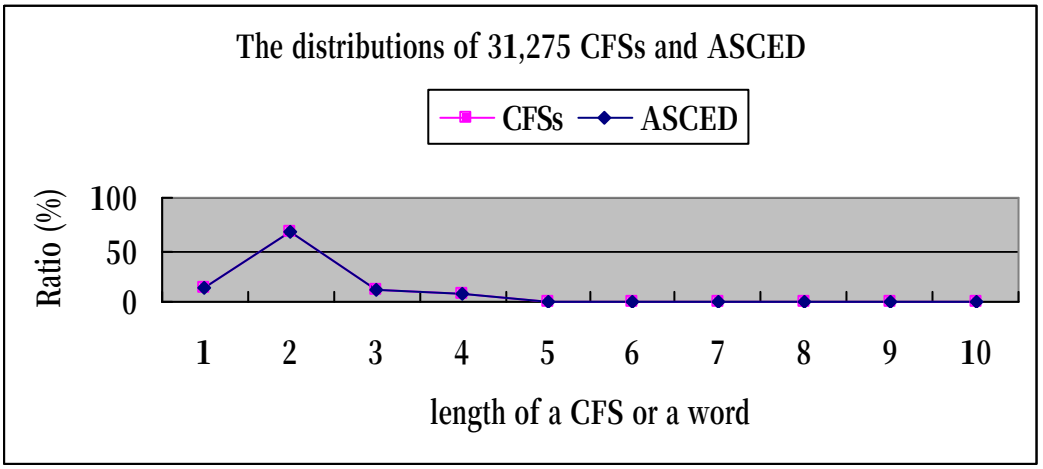

Figure 3. The distributions of length of 31,275 CFSs and ASCED

Table 2. Rules used to construct the confusing set based on the Cang-Jie Chinese input method

Length of Cang-Jie code of the target character *t*	Each character *s* satisfying the conditions below is a similar character of *t*
1	The characters where Cang-Jie codes are the same as the target character.
2	A. The length of the Cang-Jie code of *s* is 2. And the length of the LCS of *s* and *t* is 1. B. The length of the Cang-Jie code of *s* is 3. And the length of the LCS of *s* and *t* is 2.
3	The length of the Cang-Jie code of *s* is 2, 3, or 4. And the length of the LCS of *s* and *t* is 2.
4	The length of Cang-Jie code of *s* is 3, 4, or 5. And the length of the LCS of *s* and *t* is 3.
5	The length of Cang-Jie code of *s* is 4. And the length of the LCS of *s* and *t* is 4.

Table 3. The precision rate of our Chinese SEC by using CFS-based unigram LM

Top n	Precision rate
1	87.32%
2	90.82%
3	92.66%
4	93.98%
5	94.98%

Table 4. Some examples where the second choice is the correct answer

Input sentence	Top 1 choice	Top 2 choice	Correct sentence
首要作的第一件工作	首長作的第一件工作	首要作的第一件工作	首要作的第一件工作
對六十歲以上的讀者	對六十歲以上的讀者	對四十歲以上的讀者	對四十歲以上的讀者
形式別致古典	形式別致古典	形式別致古樸	形式別致古樸
昨天晚間的比賽五局卡半	昨天晚間的比賽五局下半	昨天晚間的比賽五局上半	昨天晚間的比賽五局上半

Table 5. The precision rate of the Chinese SEC by using ASCDE with bigram word LM

Top n	Precision rate
1	80.95%
2	82.58%
3	83.31%
4	83.77%
5	84.09%