Research and Implementation of Chinese Word Segmentation Method Based on Maximum Probability

Zhou Shiying

College of Plant Science, Jilin University, Changchun, China

964882432@qq.coml

Keywords: Chinese word segmentation, maximum probability method, statistical model

Abstract: With the rapid growth of network resources, how to extract comprehensive and accurate information from these massive network information has become increasingly important. In this paper, the method of Chinese word segmentation in full-text retrieval system is studied, and a Chinese word segmentation method based on maximum probability method is proposed. The experimental data show that the algorithm can effectively solve the problem of Chinese word segmentation.

1. Introduction

Chinese Words differ from Western languages in writing habits. Western languages are separated by spaces between words, while Chinese words are the basic units of language, so word segmentation has become a necessary step for automatic Chinese analysis.Whether word segmentation is done manually or by a computer, there must be a standard or specification to show how word segmentation is correct.There are two major problems in Chinese word segmentation: ambiguity and word recognition. Chinese automatic word segmentation can be roughly divided into two kinds: probabilistic method and non-probabilistic method.

Non-probabilistic methods are based on string matching, mainly including maximum matching method, minimum matching method, forward scanning method, reverse scanning method, adding matching method and subtracting matching method. The maximum matching method is more practical.

Probabilistic method is to build an automatic word segmentation statistical model, get the parameters of each group of the model, and then select the most probabilistic word string from all possible word strings as the output result. Generally, automatic word segmentation requires a vocabulary. The size of the vocabulary determines the effectiveness of the algorithm to some extent.

2. Maximum Probability Method

2.1. Noise Channel Model

Suppose there is a noise channel with an input and an output terminal, and the input terminal has a signal sequence I, which is transmitted to the output through a channel, and its signal sequence O. Owing to the noise in the channel, usually I is not equal to O. According to Bayesian formula, O can be restored to I:

$$P(I_i \mid O) = \frac{P(I_i)P(O \mid I_i)}{\sum_{j=1}^{n} P(I_j)P(O \mid I_j)} \tag{1}$$

In principle, we can find out I_j all corresponding to O, and get the most probabilistic rate I' by comparison. Since denominator does not work in comparison, it can be simplified as follows:

$$I' = \arg\max_{I} P(I \mid O) = \arg\max_{I} P(I)P(O \mid I) \tag{2}$$

This formula indicates that in order to find such a parameter I, it can make the $P(I)P(O|I)$ maximum possible. This is the so-called noise channel model.

2.2. Markov Process and N-gram

A symbol string $w_1, w_2, ..., w_n$, If the probability of each occurrence of w_i is only related to the j symbols appearing in front of it, then the process of symbol change is called the J-order Markov process.

Consider the simplest first-order Markov process in which each symbol is related to only one symbol that appears before, then $P(w_1, w_2, ..., w_n) = P(w_1)P(w_2 | w_1)....P(w_n | w_{n-1})$. If assumed w_0, it can be expressed more simply as follows:

$$P(w_1, w_2, ..., w_n) = \prod_{i=1}^{n} P(w_i | w_{i-1}) \tag{3}$$

$\prod_{i=1}^{n} P(w_i | w_{i-1})$ represents the product of n terms.

Similarly, in the second-order Markov process, a symbol is only related to the two symbols that appear in front of it. The probability of a symbol string is:

$$P(w_1, w_2, ..., w_n) = \prod_{i=1}^{n} P(w_i | w_{i-1}w_{i-2}) \tag{4}$$

In this way, we transform the probability calculation of the symbol string into the conditional probability calculation of each symbol in the string. A concept associated with Markov processes is N-gram. N-gram means that the conditional probability of the current symbol depends on the transition probability from the previous N-1 symbol to it. Therefore, ternary grammar corresponds to second-order Markov process, binary grammar corresponds to first-order Markov process, and N-gram corresponds to N-1 order Markov process. If we think that the probability of a string of symbols is the product of the probability of each symbol, it is the unary grammar.

2.3. Statistical Model of Chinese Word Segmentation

According to the noise channel model, we can think that a word string is transmitted through the channel. Because of the noise interference, the word boundary marker is lost and a Chinese character string is transformed into a Chinese character string at the output. So word segmentation is a string of Chinese characters that has the maximum probability corresponding to it. As follows:

$$W' = \arg\max_{W} P(W | Z) \tag{5}$$

According to the Bayesian formula:

$$W' = \arg\max_{W} P(W | Z) = \arg\max_{W} \frac{P(W)P(Z | W)}{P(Z)} \tag{6}$$

Among them, $P(Z)$ is the probability of Chinese character strings ,it is the same for all candidate word strings. $P(Z | W)$ is the conditional probability of word strings to Chinese character strings. Obviously, under the condition of known word strings, the probability of corresponding Chinese character strings is 1, which need not be considered. Just consider the probability $P(W)$ of word strings. Therefore, the above formula can be simplified as follows:

$$W' = \arg\max_{W} P(W) \tag{7}$$

That is to say, the most probable word string is the best one. The string probability can be calculated by the N-gram mentioned above. If binary grammar is used, then:

$$P(W) = \prod_{i=1}^{n} p(w_i \mid w_{i-1}) \tag{8}$$

Among them, $1 \le i \le n$, w_0 is a fictitious prefix.

3. Algorithmic Implementation

If we use binary grammar as analyzed above, there is a problem that the probability matrix of word-to-word transition will be very large. Therefore, this paper uses unary grammar to calculate the probability of word string, that is:

$$P(W) = \prod_{i=1}^{n} P(W_i) \tag{9}$$

The probability of each word can be estimated by its frequency. This paper uses the vocabulary statistics of Peking University Computing Language. It is a dictionary obtained from the People's Daily Corpus. It contains more than 100,000 entries and frequency information, and the frequency information of Chinese characters used as surnames and names. The size of the corpus is about 200,000 words. The probability of each word is obtained by dividing the frequency of the word itself by the size of the vocabulary. If a Chinese character is not found in this table, the frequency of occurrence is set to be 1.

The basic idea of this paper is to find out all possible words in the input string according to the vocabulary, then find all possible segmentation paths , and find the best path from these paths as the output result.

3.1. Converting Probability into Cost

Because the probability of each word is a very small positive number, if the Chinese character string is long, the probability of the final possible word string is close to 0, or even can not be expressed on the machine, of course, it can not be compared. In this paper, a more commonly used method is to find the sum of logarithms of each word and turn multiplication into addition. The probabilities of words are all positive numbers less than 1, which are negative to the value and positive to the contrary. This positive number serves as the "cost" of the word, and the corresponding "cost" of the word string is the sum of the "costs" of each word, as follows:

$$Fee(W) = \sum_{i=1}^{i=n} -\log(P(W_i)) \tag{10}$$

3.2. Algorithmic Design

If the starting word A passes through the word P and the ending word G arrives at the end word G, then the sub-path from the starting word P to the ending word G through the word H must also be the shortest path for all possible different paths from the starting word P to the ending word G. That is, the global best path must be the local best path, but the local best path is not necessarily the global best path. This characteristic conforms to the optimality principle of dynamic programming - whatever the initial state of the process and the initial decision, the rest of the decisions must form an optimal decision sequence relative to the state produced by the initial decision. In this way, the best path can be obtained by the following recursive formula:

$$dp[w_1] = Fee(w_1) \tag{11}$$

$$dp[w_i] = \min(dp[w_j] + Fee(w_i)) \tag{12}$$

Among them, wj is the word next to wi, and dp[wi] is the cost synthesis from word string to word wi.

Then the optimal path is the path corresponding to dp[wn].

The algorithm implementation process is shown in Figure 1.

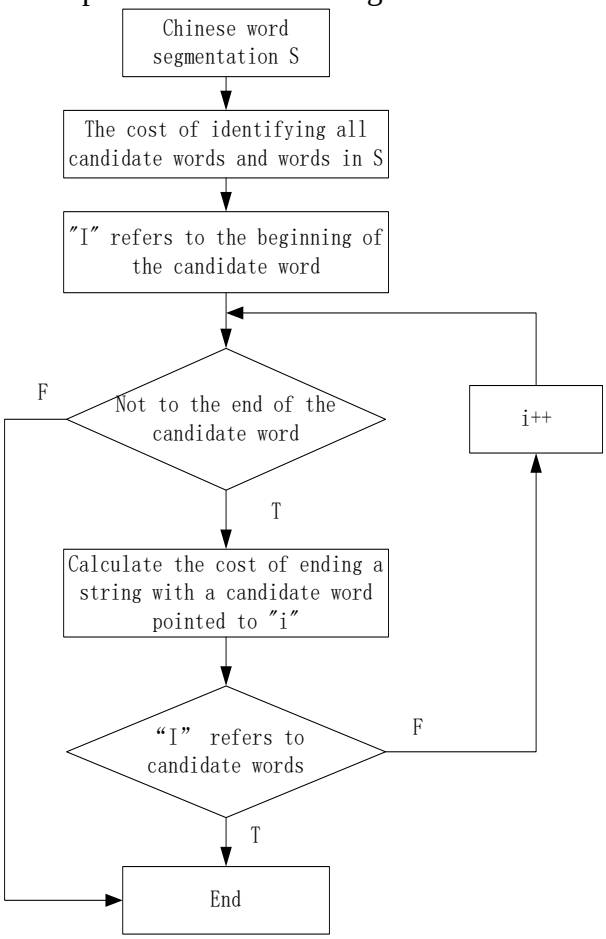

Figure 1 Word Segmentation Function Flow Chart.

3.3. Analysis of experimental results

To test the function of word segmentation in this paper, a paragraph of text will be processed by word segmentation.

We have separated 35 words from 68 Chinese characters and 28 meaningful words, of which 25 are correct words. Accuracy of Chinese Word Segmentation:

Correct Vocabulary/Total Meaningful Word Segmentation×100% = 25/28×100%=89.28%.

4. Conclusion

Firstly, this paper discusses the principle of Chinese word segmentation and several common word segmentation methods. Then, based on the basic idea of maximum probability method, an algorithm for Chinese word segmentation is proposed. The method is implemented using the thesaurus provided by Peking University Computing Language as the source of database. The experimental results show that the method is effective.

References

[1] Guohong Fu, Chunyu Kit, Jonathan J. Webster, “Chinese word segmentation as morpheme-based lexical chunking” . Information Sciences . 2008 (9),pp.2282-2296.

[2] Fei Li, Meishan Zhang, Guohong Fu, Donghong Ji, A neural joint model for entity and relation extraction from biomedical tex, BMC Bioinformatics. 2017(1), pp. 271–350.

[3] Meishan Zhang, Nan Yu, Guohong Fu, A Simple and Effective Neural Model for Joint Word Segmentation and POS Tagging, IEEE/ACM Transactions on Audio, Speech and Language

Processing (TASLP). 2018(9), pp.1528–1538.

[4] Turian J,Ratinov L,Bengio Y,Word representations: a simple and general method for semi-supervised learning,Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics . 2010,pp.2515-2521

[5] Hai Zhao, Chunyu Kit,Integrating unsupervised and supervised word segmentation,The role of goodness measures. Information Sciences. 2010(1),pp.163-183