Chinese Word Segmentation based on Maximum Matching and Word Binding Force

Pak-kwong Wong and Chorkin Chan
Department of Computer Science
The University of Hong Kong
Pokfulam Road
Hong Kong
pkwong@cs.hku.hk and cchan@cs.hku.hk

Abstract

A Chinese word segmentation algorithm based on forward maximum matching and word binding force is proposed in this paper. This algorithm plays a key role in post-processing the output of a character or speech recognizer in determining the proper word sequence corresponding to an input line of character images or a speech waveform. To support this algorithm, a text corpus of over 63 millions characters is employed to enrich an 80,000-words lexicon in terms of its word entries and word binding forces. As it stands now, given an input line of text, the word segmentor can process on the average 210,000 characters per second when running on an IBM RISC System/6000 3BT workstation with a correct word identification rate of 99.74%.

1 Introduction

A language model as a post-processor is essential to a recognizer of speech or characters in order to determine the appropriate word sequence and hence the semantics of an input line of text or utterance. It is well known that an N-gram statistics language model is just as effective as, but much more efficient than, a syntactic/semantic analyser in determining the correct word sequence. A necessary condition to successful collection of N-gram statistics is the existence of a comprehensive lexicon and a large text corpus. The latter must be lexically analysed in order to identify all the words, from which, N-gram statistics can be derived.

About 5,000 characters are being used in modern Chinese and they are the building blocks of all words. Almost every character is a word and most words are of one or two characters long but there are also abundant words longer than two characters. Before it is segmented into words, a line of text is just a sequence of characters and there are numerous word segmentation alternatives. Usually, all but one of these alternatives are syntactically and/or semantically incorrect. This is the case because unlike texts in English, Chinese texts have no word markers. A first step towards building a language model based on N-gram statistics is to develop an efficient lexical analyser to identify all the words in the corpus.

Word segmentation algorithms belong to one of two types in general, viz., the structural (Wang et al., 1991) and the statistical type (Lua, 1990)(Lua and Gan, 1994)(Sproat and Shih, 1990) respectively. A structural algorithm resolves segmentation ambiguities by examining the structural relationships between words, while a statistical algorithm compares the usage frequencies of the words and their ordered combinations instead. Both approaches have serious limitations.

2 Maximum Matching Method for Segmentation

Maximum matching (Liu et al., 1994) is one of the most popular structural segmentation algorithms for Chinese texts. This method favours long words and is a greedy algorithm by design, hence, suboptimal. Segmentation may start from either end of the line without any difference in segmentation results. In this paper, the forward direction is adopted. The major advantage of maximum matching is its efficiency while its segmentation accuracy can be expected to lie around 95%.

3 Word Frequency Method for Segmentation

In this statistical approach in terms of word frequencies, a lexicon needs not only a rich repertoire of word entries, but also the usage frequency of each word. To segment a line of text, each possible segmentation alternative is evaluated according to the product of the word frequencies of the words segmented. The word sequence with the highest frequency product is accepted as correct. This method is simple but its accuracy depends heavily on the accuracy of the usage frequencies. The usage frequency of a word differs greatly from one type of documents to another, say, a passage of world news as against a technical report. Since there are tens of thousands of words actively used, one needs a gigantic collection of texts to make an accurate estimate, but by then, the estimate is just an average and it may not be suitable for any type of document at all. In other words, the variance of such an estimate is too great making the estimate useless.

4 The Lexicon

Most Chinese linguists accept the definition of a word as the minimum unit that is semantically complete and can be put together as building blocks to form a sentence. However, in Chinese, words can be united to form compound words, and they in turn, can combine further to form yet higher ordered compound words. As a matter of fact, compound words are extremely common and they exist in large numbers. It is impossible to include all compound words into the lexicon but just to keep those which are frequently used and have the word components united closely. A lexicon was acquired from the Institute of Information Science, Academia Sinica in Taiwan. There are 78410 word entries in this lexicon, each associated with a usage frequency. A corpus of over 63 million characters of news lines was acquired from China. Due to cultural differences of the two societies, there are many words encountered in the corpus but not in the lexicon. The latter must therefore be enriched before it can be applied to perform the lexical analysis. The first step towards this end is to merge a lexicon published in China into this one, increasing the number of word entries to 85,855.

5 The Proposed Word Segmentation Algorithm

The proposed algorithm of this paper makes use of a forward maximum matching strategy to identify words. In this respect, this algorithm is a structural approach. Under this strategy, errors are usually associated with single-character words. If the first character of a line is identified as a single-character word, what it means is that there is no multi-character word entry in the lexicon that starts with such a character. In that case, there is not much one can do about it. On the other hand, when a character is identified as a single-character word β following another word α in the line, one cannot help wondering whether the sole character composing β should not be combined with the suffix of α to form another word instead, even if that means changing α into a shorter word. In that case, every possible word sequence alternative corresponding to the sub-sequence of characters from α and β together will be evaluated according to the product of its constituent word binding forces. The binding force of a word is a measure of how strongly the characters composing the word are bound together as a single unit. This force is often equated to the usage frequency of the word. In this respect, the proposed algorithm is a statistical approach. It is as efficient as the maximum matching method because word binding forces are utilized only in exceptional cases. However, much of the word ambiguities are eliminated, leading to a very high word identification accuracy. Segmentation errors associated with multi-character words can be reduced by adding or deleting words to or from the lexicon as well as adjusting word binding forces.

6 Structure of the Lexicon

Words in the lexicon are divided into 5 groups according to word lengths. They correspond to words of 1, 2, 3, 4, and more than 4 characters with group sizes equal to 7025, 53532, 12939, 11269, and 1090 respectively. Since most of the time spent in analyzing a line of text is in finding a match among the lexicon entries, a clever organization of the lexicon speeds up the searching process tremendously. Most Chinese words are of one or two characters only. Searching for longer words before shorter ones as practised in maximum matching means spending a great deal of time searching for non-existent targets. To overcome this problem, the following measures are taken to organize the lexicon for fast search:

- All single character words are stored in a table of 32768 bins. Since the internal code of a character takes 2 bytes, bits 1-15 are used as the bin address for the word.
- All 2-character words are stored in a separate table of 65536 bins. The two low order bytes of the two characters are used as a short integer for bin address. Should there be other words contesting for the same bin, they are kept in a linked list.
- Any 3-character word is split into a 2-character prefix and a 1-character suffix. The prefix will be stored in the bin table for 2-character words with clear indication of its prefix status. The suffix will be stored in the bin table for 1-character words, again, with clear indication of its suffix status. All duplicate entries are combined, i.e., if α is a word as well as a suffix, the two entries are combined into one with an indication that it can serve as a word as well as a suffix.
- Any 4-character word is divided up into a 2-character prefix and a 2-character suffix, both stored in the bin table for 2-character words, with clear indications of their respective status. Each prefix points to a linked list of associated suffixes.

- Any word longer than 4 characters will be divided into a 2-character prefix, a 2-character infix and a suffix. The prefix and the infix are stored in the bin table for 2-character words, with clear indications of their status. Each prefix points to a linked list of associated infixes and each infix in turn, points to a linked list of associated suffixes.

Maximum matching segmentation of a sequence of characters "...abcdefghij..." at the character "a" starts with matching "ab" against the 2-character words table. If no match is found, then, "a" is assumed a 1-character word and maximum matching moves on to "b". If a match is found, then, "ab" is investigated to see if it can be a prefix. If it cannot, then "ab" is a 2-character word and maximum matching moves on to "c". If it can, then one examines if it can be associated with an infix. If it can, then one examines if "cd" can be an infix associated with "ab". If the answer is negative, then the possibility of "abcd" being a word is considered. If that fails again, then "c" in the table of 1-character words is examined to see if it can be a suffix. If it can, then "abc" will be examined to see if can be a word by searching the 1-character suffix linked list pointed at by "ab". Otherwise, one has to accept that "ab" is a 2-character word and moves on to start matching at "c". If "cd" can be an infix preceded by "ab", the linked list pointed at by "cd" as an infix will be searched for the longest possible suffix to combine with "abcd" as its prefix. If no match can be found, then one has to give up "cd" as an infix to "ab".

7 Training of the System

Despite the fact that the lexicon acquired from Taiwan has been augmented with words from another lexicon developed in China, when it is applied to segment 1.2 million character news passages in blocks of 10,000 characters each randomly selected over the text corpus, an average word segmentation error rate (μ) of 2.51% was found with a standard deviation (σ) of 0.57%, mostly caused by uncommon words not included in the enriched lexicon. Then it is decided that the lexicon should be further enriched with new words and adjusted word binding forces over a number of generations. In generation i, n new blocks of text are picked randomly from the corpus and words segmented using the lexicon enriched in the previous generation. This process will stop when μ levels off over several generations. The $100(1-\alpha)\%$ confidence interval of μ in generation i is $\pm t_{0.5\alpha,n-1}\sigma/\sqrt{n}$ where σ is the standard deviation of error rates in generation $i-1$, and n is the number of blocks to be segmented in generation i. $t_{0.5\alpha,n-1}$ is the density function of $(0.5\alpha, n-1)$ degrees of freedom(Devore, 1991). Throughout the experiments below, n is always chosen to be 20 so that the 90% confidence interval (i.e., $\alpha = 0.1$) of μ is about $\pm 0.23\%$.

8 Experimental Results

The lexicon has been updated over six generations after being applied to word segment 1.2 million characters. The vocabulary increases from 85855 words to 87326 words. The segmentation error rates over seven generations of the training process are shown in the table below:

Lexicon Generation Number	Error Rate μ over a text of 200,000 Characters	
	Max. Mat.	Max. Mat. & Word Bind. Force
0	5.71%	2.32%
1	5.20%	2.16%
2	4.66%	1.88%
3	4.98%	1.62%
4	2.60%	0.43%
5	2.47%	0.30%
6	2.44%	0.26%

Most of these errors occur in proper nouns not included in the lexicon. They are hard to avoid unless they become popular enough to be added to the lexicon. The CPU time used for segmenting a text of 1,200,000 characters is 5.7 seconds on an IBM RISC System/6000 3BT computer.

9 Conclusion

Lexical analysis is a basic process of analyzing and understanding a language. The proposed algorithm provides a highly accurate and highly efficient way for word segmentation of Chinese texts. Due to cultural differences, the same language used in different geographical regions and different applications can be quite different causing problems in lexical analysis. However, by introducing new words into and adjusting word binding forces in the lexicon, such difficulties can be greatly mitigated.

This word segmentor will be applied to word segment the entire corpus of 63 million characters before N-gram statistics will be collected for post-processing recognizer outputs.

References

Jay L. Devore. 1991. Probability and Statistics for Engineering and Sciences. *Duxbury Press*, pages 272-276.

Yuan Liu, Qiang Tan, and Kun Xu Shen. 1994. The Word Segmentation Rules and Automatic Word Segmentation Methods for Chinese Information Processing (in Chinese). *Qing Hua University Press and Guang Xi Science and Technology Press*, page 36.

Kim-Teng Lua and Kok-Wee Gan. 1994. An Application of Information Theory in Chinese Word Segmentation. *Computer Processing of Chinese and Oriental Languages*, Vol. 8, No. 1, pages 115–123, June.

K.T. Lua. 1990. From Character to Word – An Application of Information Theory. *Computer Processing of Chinese and Oriental Languages*, Vol. 4, No. 4, pages 304–313, March.

Liang-Jyh Wang, Tzusheng Pei, Wei-Chuan Li, and Lih-Ching R. Huang. 1991. A Parsing Method for Identifying Words in Mandarin Chinese Sentences. In *Processings of 12th International Joint Conference on Artificial Intelligence*, pages 1018–1023, Darling Harbour, Sydney, Australia, 24-30 August.

Richard Sproat and Chilin Shih. 1990. A Statistical Method for Finding Word Boundaries in Chinese Text. *Computer Processing of Chinese and Oriental Languages*, Vol. 4, No. 4, pages 336–349, March.