The Dictionary Mechanism for Chinese Word Segmentation

---Hashing the Initial and Termination of the Chinese Words

MIAO Liming
Mathematics and Information Technology
Hanshan Normal University
Guangdong,China

***Abstract*—For the purpose of improving the efficiency of Chinese word segmentation, this paper puted forward the method of using initial and termination to distinguish the Chinese words on the basis of analyzing Chinese word characteristics in Chinese dictionary. The result showed that efficiency is enhanced by adopting this mechanism.**

Keywords-segmentation dictionary; hash conflict; distinction degree; segmentation algorithm

I. INTRODUCTION

Chinese word segmentation is widely used in the Chinese information processing such as machine translation, information retrieval, intelligence analysis and expert systems etc. At present, Chinese word segmentation system which is widely used is based on the dictionary. Its properties are dependent on the algorithm of word segmentation and the structure of dictionary etc. The paper analyzed and compared the existing dictionary mechanism in the first part , then counted and analyzed the characteristics of the Chinese words, and proposed an improved mechanism for Chinese word segmentation dictionary.

II. COMPARISON OF EXISTING DICTIONARY MECHANISM

The paper take the experimental study, and then compared the binary-seek-by-word , the TRIE indexing tree and the binary-seek-by-characters. The conclusion shows that the TRIE indexing tree and the binary-seek-by-characters is similar in terms of time efficiency, both of them are more excellent than the binary-seek-by-word. And the dictionary mechanism of the binary-seek-by-characters has been greatly improved in space efficiency [1]. Reference [2] proposed a method of long term priority by character tree storage, which matched the long term priority by verbatim. It was better than the TRIE indexing tree mechanism in matching accuracy. Rreference [3] proposed the mechanism of double-character-hash-indexing. This mechanism established the sequential hash table for the first and the second word on the terms. It matched the two words previous by verbatim, and the remaining by binary-seek-by-word. The processing speed was much faster than the TRIE indexing tree and the binary-seek-by-characters. Reference [4] used the concept of automatic machines to build the dictionary, and the dictionary was base on a single array. When matching a string, it needs to match the transfer conditions for each character. And when each word has been set a match for 4 times , the transferation will be completed. Reference [5] presented the dictionary structure in accordance with the character number of words to classification.It means that words in the same prefix should be permutated by the number of the charatcers it has. This segmentation algorithm can reduce the matching number of the Chinese word segmentation process. And the efficient of Chinese word segmentation is faster than the binary-seek-by-characters and the TRIE indexing tree. Reference [6] processed the structure of words by stratification, and stored different words by different ways. It takes the method of binary-seek-by-characters for matching, which efficiency is less than the mechanism of TRIE indexing tree.

In conclusion, the TRIE indexing tree and the binary-seek-by-characters dictionary mechanism both has fine efficiency in time and space. The mechanism efficiency of Chinese word segmentation in reference [3] and reference [5] are both better than the TRIE indexing tree and the binary-seek-by-characters.

III. STATISTICS ANALYSIS

In this paper, the dictionary is from the On-line Chinese Tools. It has a total of 119,802 Chinese words, of which 72,965 are two-character words, 19,341 are three-character words, 25,853 are four-character words and 1,643 are four-character-or-more words. We took a statistics of Chinese words in dictionary, which statistics is in accordance with the character number of Chinese words. The statistical items are that initial of Chinese Words is the same, the first and the second character of Chinese words are the same and the initial and termination are the same. The results are in TABLE I.

TABLE I. Classification Statistics

Characters Number	Same Initial		Same the First and the Second Character		Same Initial and Termination	
	max_i	D(%)	max_{fs}	D(%)	max_{it}	D(%)
Two-character Words	473	1.2	1	100.0	1	100.0
Three-character Words	559	3.1	24	53.6	20	71.2
Four-character Words	501	2.8	49	47.9	19	73.7
Other Words	60	20.6	13	65.0	10	73.2

Below is the description of the various items in TABLE I.

978-0-7695-4304-8/10 $26.00 © 2010 IEEE
DOI 10.1109/GCIS.2010.193

A. max_i

The max_i is the largest number of words in a group of which initial of the words is the same. It means that max_i is hash conflict maximum of same initial. For example, it is most that initial of words is "bu" in two-character words, which number is 473, and then max_i =473 for two-character words.

B. max_{fs}

The max_{fs} is the largest number of words in a group of which the first and the second character are the same.

C. max_{it}

The max_{it} is the largest number of words in a group of which the initial and termination of the words are the same.

D. D—Word Distinction Degree

If the total number of Chinese words is N and the number of distinguished Chinese words is M under restricting conditions, then the word distinction degree is D = M / N under the conditions. The high word distinction degree, the more number of words can be distinguished directly.

TABLE I results showed that: The word distinction degree is the lowest, when the initial of words is the same. The word distinction degree is 2.1% after classified and subtotaled by the character number of words, and it can distinguish 2,544 words.

When the first and the second character are the same, the word distinction degree is greatly improved. The word distinction degree is 100% for two-character words. The word distinction degree is 50.9% except for two-character words after classified and subtotaled by the character number of words. It can distinguish 2,544 words.

When the distinction condition is that initial and termination is the same, the situation of two-character words is the same as the situation which the first and the second character is the same. The word distinction degree is obviously improved for three-character words and four-character words, which is 17.6% and 25.8%. The word distinction degree is 72.6% except for two-character words after classified and subtotaled by the character number of words. It can distinguish 34,024 words. If the number of words includes the two-character words, it can distinguish 106,989 words. Thus, there are only 12,813 words of hash conflict.

IV. THE DICTIONARY MECHANISM OF HASH FOR INITIAL AND TERMINATION

The statistical results in TABLE I showed that the higher degree of word distinction degree, the more number of words can be identified, and the less hash conflict. If we take the word segmentation algorithm which word distinction degree is high, then the number of matcheing can be reduced and the efficiency of the algorithm can be improved. The word distinction degree is proportional to the number of constraint condition. Because the more constraint condition would increase the complexity of segmentation algorithm, so we should choose an appropriate constraint condition. In this paper, we take the distinction condition which the initial and termination is the same, classification and organization of words according to the number of character, and we deal the words for a hash conflict with open addressing method.

A. The Structure of Word Segmentation Dictionary

The dictionary text is arranged in ascending order by alphabet. It is divided into four levels when the dictionary text is loaded into memory: the initial hash table, index table of words, termination hash table and the words table.

B. Building Dictionary of Word Segmentation

Below is the algorithm which use the array to build dictionary of word segmentation.

procedure＜load dictionary＞
declare＜dic＞as＜array＞
read a words from dictionary files
loop while < no dictionary file end >
cut the initial and termination
loop until < the initial without same >
the same initial and termination initialization
save the words to array
count characters of the longest words of the same initial and termination
read next words and cut the termination
end loop < the initial without same >
end loop < no dictionary file end >
return segmentation dictionary
end＜load dictionary＞

Below is the code of loading dictionary, which use PHP to encode, and Chinese character encoding is UTF-8.

```
$pointer_file=fopen('word03.txt','a+');
$dic=array(array());
$pos_first="";  $pos_termi="";
while(!feof($pointer_file))
{ $str=trim(fgets($pointer_file));
$pos_first=mb_substr($str,0,1);
$pos_termi=mb_substr($str,-1,1);
if(empty($dic["$pos_first".mb_strlen($str)."$pos_termi"][0])) {$dic["$pos_first".mb_strlen($str)."$pos_termi"][0]=0; }
$dic["$pos_first".mb_strlen($str)."$pos_termi"][0]=$dic["$pos_first".mb_strlen($str)."$pos_termi"][0]+1;
$dic["$pos_first".mb_strlen($str)."$pos_termi"][]=$str;
if($dic["$pos_first"][0]<mb_strlen($str))
{$dic["$pos_first"][0]=mb_strlen($str);}
if($dic["$pos_termi"][0]<mb_strlen($str))
{$dic["$pos_termi"][0]=mb_strlen($str);}}
fclose($pointer_file); return $dic;
```

C. Word Segmentation Algorithm

1) Single-character word processing

This segmentation dictionary contains no single-character words. When word segmentation, the single-character words are not matched, and directly segmented to form a string of single-character words. If the single-character words can form a multi-words, then it is to seek the longest words. If the single-character words does not form a multi-words, then it is a string of single-character words.

2) Two-character word processing

Two-character words are all mapped conflict-free. The matching method are divided into two situations: the first situation is only existing two-character words, and the second situation is that words contain more than two characters. The former can be segmented into a two-character words, and the latter can be segmented a multi-words by the maximum number of character in words.

3) Multi-words Processing

The multi-words processing is divided into two situations which have the hash conflict and the hash conflict-free. The words of hash conflict-free can be segmented directly, and the words of hash conflict can be matched by binary-seek-by-word.

Below is the algorithm of the maximum matching method.

```
procedure＜word segmentation＞
cut the initial of words
if< existing the initial words >
Y: if <only existing two-character words >
      Y: cut two character of the words
      N: cut the longest words
   end if<only existing two-character words >
N: cut and build the single-character words
end if< existing the initial words >
loop until <the string length >2>
cut the termination
loop until < has the same length, initial is same to termination >
seek by word
end loop< has the same length>
deal with matching words
end loop< the string length >2>
does not constitute a two-character words, build a string of single-character words
end＜word segmentation＞
```

Below is the code of hash conflict part, which use PHP to encode.

```
if(!empty($loaded_dic["$pos_first"][0]))
{ if($loaded_dic["$pos_first"][0]==2)
{ $buf=trim(mb_substr($buf,0,2));
}else{
$buf=mb_substr($buf,0,$loaded_dic["$pos_first"][0]);
$pos_termi=mb_substr($buf,-1,1);
while(($loaded_dic["$pos_first".mb_strlen($buf)."$pos_termi"][0]<1)&(mb_strlen($buf)>2))
{ $buf=mb_substr($buf,0,mb_strlen($buf)-1);
$pos_termi=mb_substr($buf,-1,1);                }}else{
$buf=$pos_first; } $key_exit=0;
while(mb_strlen($buf)>2)
{$k="$pos_first".mb_strlen($buf)."$pos_termi";
 if($loaded_dic[$k][1]==$buf)
{break;}
while(!empty($loaded_dic[$k][$i]))
{if($buf==$loaded_dic[$k][$i])
{ $key_exit=1; break;} $i=$i+1;}
if($key_exit==1){ break;}
$buf=trim(mb_substr($buf,0,mb_strlen($buf)-1));
$pos_termi=mb_substr($buf,-1,1);}
if(mb_strlen($buf)==2)
{if($loaded_dic["$pos_first".'2'.mb_substr($buf,-1,1)][0]!=1){$buf=mb_substr($buf,0,1); }}
```

V. EXPERIMENTAL TESTING

We take the experiment to test the segmentation algorithm for reference [3], reference [5] and this paper. we compare the word segmentation efficiency of time and space. In order to ensure the accuracy of test, we take the same environment of hardware and software, which we use the PHP to encode and use a array to build the words table. The word segmentation approach is the maximum matching method. The binary-seek is used to the double-character-hash-indexing, and the seeking by words is used to the other two mechanisms are to use. The test dictionary is a total of 119,802 Chinese words (not including single-character words), and the test text is extracted from a variety of news, IT, science, medicine, etc, a total of 322k.

After several tests, the result showed that the occupation memory of the three kind of dictionary mechanisms are similar, and the dictionary mechanism of the initial and termination hash is most. Its space efficiency is lowest, and is 94k and 8k respectively more than the other two methods. In time efficiency, the dictionary mechanism of the initial and termination hash has a better segmentation speed, and faster than the other two mechanisms, 8.7% and 3.7% respectively.

REFERENCES

[1] SUN Maosong, ZUO Zhengping and HUANG Changning, “An experimental study on dictionary mechanism for Chinese word segmentation,” Journal of Chinese Information Processing, vol. 14, 2000, pp. 1–6.

[2] FU Liyun and LIU Xin, “The improved algorithm of Chinese word segmentation based of dictionary,” Journal of Information, Jan. 2006, pp. 42–43.

[3] LI Qinghu, CHEN Yujian and SUN Jiaguang, “A new dictionary mechanism for Chinese word segmentation,” Journal of Chinese Information Processing, vol. 17, 2003, pp. 13–18.

[4] WEI Jin and CHANG Chaowen, “Full-mapping dictionary implemented by single array,”Computer Engineering and Applications, vol. 43, 2007, pp. 184-186.

[5] GUO Yi, “An improved mechanism on the Chinese word segmentation,” Computer Knowledge and Technology, vol. 7, 2008, pp. 1240–1245.

[6] Zhou Chengyuan, Zhu Min and Yang Yun, “Research on Chinese word segmentation algorithm based on the Dictionary,”Computer and Digital Engineering, vol. 37, 2009, pp. 69–71.

[7] HOU Min, Computational Linguistics and Chinese Automatic Analysis, 1st ed., Beijing: Broadcasting Institute, 1999, pp. 101–102.