Research and Implementation of Unlisted Word Discovery System

Shi-wei JIA\textsuperscript{1,\textasteriskcentered} and Yu-meng ZHANG\textsuperscript{2}

\textsuperscript{1}Department of Library and Information Archives, Shanghai University, Shanghai, China
\textsuperscript{2}Business School of Ningbo University, Ningbo, China
\textsuperscript{\textasteriskcentered}sw_jia.foxmail.com
\textsuperscript{\textasteriskcentered}Corresponding author

Keywords: Unlisted Word, Apriori Algorithm, Transaction Compression.

Abstract. Unlisted word is a problem in Chinese word segmentation. In this paper, an improved Apriori algorithm is proposed, which can quickly and accurately identify unlisted words. The improved algorithm applied a compressed database approach to reduce the number of transactions. Compared with the traditional n-tuple algorithm and NApriori algorithm, it is faster and more effective.

Introduction

Automatic identification of unlisted word is an important problem in Chinese information processing. It has a wide range of applications in information retrieval, information filtering and so on [1]. With the development of society, there have been a large number of unlisted words, greatly increasing the difficulty of Chinese information processing, resulting in Chinese automatic word segmentation often mistakes. A large number of literatures show that the errors caused by unlisted words are much greater than those caused by ambiguity [2-3]. In order to solve this problem, domestic and foreign researchers put forward a variety of programs, roughly divided into three, one is based on the dictionary method, one is based on the understanding of the method, one is based on statistical methods. However, unlisted words refer to words that do not exist in the dictionary, and the dictionary-based approach can’t find unlisted words, and the method based on comprehension is still in the initial stage of the study. Therefore, the method based on statistical recognition is the current mainstream method [4-6]. For example, Nie used the method of statistic and general likelihood ratio to calculate inter-word correlation, the unlisted words were obtained automatically from corpus to segment words or construct automatically dictionary [6].

As we all know, there is no obvious delineation between the Chinese word words, which makes the current mature theory is not much, and the main research results focused on domestic scholars. For example, Professor Chen proposed a package of solutions for unlisted questions [7], Professor Liang designed the CDWS model [8], and Professor He put forward the concept of expert word segmentation system [9] and so on.

In recent years, the identification of proprietary nouns has achieved good results, the formation of a more mature word recognition system, on behalf of the Chinese Academy of Sciences Chinese word segmentation system (ICTCLAS), PanGu segmentation system and so on. For name recognition, now has been able to identify the
"Lao Zhang", "Xiao Li" and other slang, and the accuracy is high. But for the new words from network, there is not a good way to identify.

Based on the previous research, this paper proposes an improved Apriori algorithm based on the idea of data mining. The algorithm uses its anti-monotonicity to compress the number of transactions, reduce the number of candidate strings, and improve the unlisted words the efficiency and quality of identification.

Design of Unknown Word Recognition Algorithm

Corpus Construction

Corpus, also known as language database, is the basic resource for unrecognized word recognition. With the development of society and the popularization of the Internet, a large number of network new words have emerged. These network new words have caused great distress to Chinese information processing. The traditional static corpus has been unable to meet the demand of word segmentation. Therefore, it is imperative to construct corpus with Internet resources.

In China, the portal is the main place where network new words appear. In order to construct a comprehensive and high-quality corpus, the average daily PV, update frequency and so on of the major domestic portal websites were analyzed and finally decided to adopt Sina as the information source. Use crawler technology to grab a large number of web pages, and then parse the page content to build a clean corpus.

Text Preprocessing

In order to improve the speed of algorithm recognition, it should be as much as possible to cut the text into a shorter string. As the corpus resources from the Internet, the format is not standardized, so the text preprocessing is a very critical step. Although there is no clear delineation between Chinese words and words, there are two types of separators that can be used: (1) non-Chinese characters, including punctuation, numbers, letters, etc. (2) noise words, refers to the word structure of the poor words or words, such as "the", "ah", such words exist very common, but rarely express document-related information. Thus, the text can be cut into short sentences set by using these separators, and unlisted words exist in these word segments.

An improved Apriori Recognition Algorithm

In the unlisted word recognition process, effectively identify statistical methods can greatly shorten the computation time and improve operational efficiency. The traditional n-gram method will produce a large number of candidate strings, making the statistics time-consuming, and Apriori algorithm performance will be reduced with the increase in the number of transactions, both are difficult to meet the requirements of massive data processing. In order to improve the efficiency of the operation, this paper proposes an improved Apriori unlisted word recognition algorithm, which can quickly and accurately identify unlisted words.

The improved algorithm will compress the number of transactions in the transaction database, perform two pruning processes at the pruning step, and mark the useless transactions to avoid scanning the next scan. This approach greatly reduces the number of transactions and reduces I/O overhead. At the same time modifies the valid frequency that the string appears in a text, to improve the accuracy of unlisted word recognition. The algorithm steps are as follows:

(1) Transaction pruning step
Scan the transaction database DB, get k-itemsets and their support numbers. According to the minimum count of support to get frequent k-itemsets Lk, and the infrequent itemsets set tag = 0, which means skip the next scan.

(2) Candidate string forming step
In order to compute Lk+1, according to the Apriori property, it is necessary to select all the connectable sets of candidate (k+1)-items that can be connected from Lk, denoted as Ck+1. Assuming that the items in the item set are sorted by word order, the connectable pair means that only two of the frequent itemsets are the last.

(3) Itemsets pruning step
According to the Apriori property, any infrequent (k-1) item set is not a subset of frequent k terms. Thus, if any (k-1) item subset of candidate k items set in Ck is not in Lk-1, the set of candidate k items can’t be frequent and can be removed from Ck.

(4) Modify valid support count step
After finding all frequent k-itemsets, correcting the number of effective support to identify the real unlisted words.

**Valid Support Count Correction**

After the statistics of the algorithm, often get some meaningless high-frequency string, these strings may be split from their parent-string. For example, "zu sai" (group match) is the sub-string of "xiao zu sai" (group stage). So they are not their independent real frequency, the support count must be modified to exclude such high-frequency string interference [10].

The valid support count of the string is equal to the frequency of its occurrence minus the frequency of its most frequently used super string, and its formula is as follows, where \( F(\cdot) \) indicates the frequency of the candidate string, and \( sup(\cdot) \) represents the frequent parent string of the candidate string:

\[
\text{Valid}(x_i) = F(x_i) - \max\{F(sup(x_i))\}
\]

**Algorithm Example**

For example, a string "ningbodaxuezainingboshijiangbeiqu, woshiningbodaxuedexuesheng." (Ningbo University in Ningbo Jiangbei District, I am a student of Ningbo University) was divided into a set \( S = \{\text{ningbodaxue}(\text{Ningbo University}), \text{ningboshijiangbeiqu}(\text{Ningbo Jiangbei District}), \text{ningbodaxue}(\text{Ningbo University}), \text{xuesheng}(\text{Student})\} \).

(1) Set the minimum count of support to 2, scan the phrase set S and count, after the first scan iteration, the candidate 1-string set \( C_1 \). And according to the minimum number of support, get frequent 1-itemsets \( L_1 = \{\text{ningbo}, \text{bod}, \text{da}, \text{xue}\} \), and all the infrequent itemsets set tag = 0, so that it will not be never scanned.

(2) Frequent 1-string set \( L_1 \) to connect, generate a new candidate 2-string set \( C_2 = \{\text{ningbo}, \text{bod}, \text{da}, \text{xue}\} \), calculate the number of each candidate string support, and finally determine the frequent 2-string set \( L_2 = \{\text{ningbo}, \text{bod}, \text{da}, \text{xue}\} \).

(3) Repeat step (2) until the k-item set can’t be found. After level-wise scanning, get frequent 3-item items and frequent 4-itemsets, as shown in Table 1.
Table 1. Support count of frequent strings.

<table>
<thead>
<tr>
<th>Strings</th>
<th>ningbo (Ningbo)</th>
<th>boda (Big waves)</th>
<th>daxue (University)</th>
<th>Ningboda (Ningbo big)</th>
<th>bodaxue (Wave University)</th>
<th>ningbodaxue (Ningbo University)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support Count</td>
<td>3</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
</tbody>
</table>

(4) the support number is effectively corrected to show the true frequency of the candidate string, as shown in Table 2.

Table 2. Valid support count of frequent strings.

<table>
<thead>
<tr>
<th>Strings</th>
<th>ningbo (Ningbo)</th>
<th>boda (Big waves)</th>
<th>daxue (University)</th>
<th>Ningboda (Ningbo big)</th>
<th>bodaxue (Wave University)</th>
<th>ningbodaxue (Ningbo University)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Valid Support Count</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>2</td>
</tr>
</tbody>
</table>

Experimental Results and Analysis

In the A6-4400M, 4G memory, Windows 7 platform. The texts with different lengths are selected randomly to text. The algorithm proposed in this paper, the algorithm in the reference [11] and the traditional n-tuple algorithm are applied to test the unlisted word extraction. The experimental results on two aspects are listed as follows: (1) the frequent string extraction efficiency; (2) the frequent string extraction accuracy.

The Efficiency of Extraction

Generate \((k + 1)\) - candidate strings on a frequent k-string basis, looping until no candidate strings are generated. The experimental results are shown in Table 3:

Table 3. Time-consuming comparison of different lengths candidate strings.

<table>
<thead>
<tr>
<th>Number of test strings (bars)</th>
<th>Number of tests (times)</th>
<th>The average time of the traditional algorithm (s)</th>
<th>The average time of the algorithm in the reference[11] (s)</th>
<th>The average time of this algorithm (s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>100</td>
<td>10</td>
<td>8.22</td>
<td>7.53</td>
<td>3.33</td>
</tr>
<tr>
<td>200</td>
<td>10</td>
<td>23.36</td>
<td>20.35</td>
<td>7.81</td>
</tr>
<tr>
<td>500</td>
<td>10</td>
<td>97.88</td>
<td>75.49</td>
<td>24.20</td>
</tr>
<tr>
<td>1000</td>
<td>10</td>
<td>249.24</td>
<td>231.58</td>
<td>55.85</td>
</tr>
<tr>
<td>2000</td>
<td>1</td>
<td>939.48</td>
<td>908.85</td>
<td>174.87</td>
</tr>
<tr>
<td>5000</td>
<td>1</td>
<td>2126.04</td>
<td>2074.94</td>
<td>426.979</td>
</tr>
</tbody>
</table>

For the same corpus, the algorithm proposed in this paper is time-consuming than the other two algorithms, and the growth rate is the most slow, in line with the time required to deal with massive data, extraction efficiency is the highest. This is due to the improvement of the Apriori algorithm, greatly compressed the number of scanning transactions, making the number of candidate string generation reduced, reducing the algorithm running time. The total number of candidate strings is shown in Figure 1. The abscissa in the graph is the length of the test text, and the ordinate is the number of candidate strings.
The Quantity and Quality of Extraction

The sample above is taken as the test object. The algorithm modifies the valid count in the final string and selects the string that satisfies the minimum supported number of conditions. The experimental results are shown in Table 4.

Table 4. The effect of frequent word extraction.

<table>
<thead>
<tr>
<th>Docid</th>
<th>Length of text</th>
<th>Number of words</th>
<th>Number of Correct words</th>
<th>Correct Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>456</td>
<td>17</td>
<td>16</td>
<td>94.1%</td>
</tr>
<tr>
<td>2</td>
<td>1035</td>
<td>19</td>
<td>18</td>
<td>94.7%</td>
</tr>
<tr>
<td>3</td>
<td>2780</td>
<td>28</td>
<td>27</td>
<td>96.4%</td>
</tr>
<tr>
<td>4</td>
<td>5170</td>
<td>98</td>
<td>95</td>
<td>96.9%</td>
</tr>
<tr>
<td>5</td>
<td>11574</td>
<td>307</td>
<td>292</td>
<td>95.1%</td>
</tr>
<tr>
<td>6</td>
<td>22535</td>
<td>650</td>
<td>618</td>
<td>95.1%</td>
</tr>
</tbody>
</table>

In order to test the validity of this algorithm, the algorithm in reference [11] and traditional  method are taken to be compared. The novel The Stewardess is taken as the test data. Set the minimum support count of 4. The experimental results are shown in Table 5.

Table 5. The comparison of different word extraction methods.

<table>
<thead>
<tr>
<th></th>
<th>The number of frequent words</th>
<th>The number of correct words</th>
<th>Correct rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Traditional Approach</td>
<td>246</td>
<td>237</td>
<td>96.34%</td>
</tr>
<tr>
<td>Approach in [11]</td>
<td>211</td>
<td>203</td>
<td>96.21%</td>
</tr>
<tr>
<td>Approach in this paper</td>
<td>225</td>
<td>217</td>
<td>96.44%</td>
</tr>
</tbody>
</table>

Acknowledgement

Fund Project: Zhejiang Provincial Department of Education Research Project (Y200907096)
References


