Statistically-Enhanced New Word Identification in a Rule-Based Chinese System

Andi Wu
Microsoft Research
One Microsoft Way
Redmond, WA 98052
Andiwu@microsoft.com

Zixin Jiang
Microsoft Research
One Microsoft Way
Redmond, WA 98052
jiangz@microsoft.com

Abstract

This paper presents a mechanism of new word identification in Chinese text where probabilities are used to filter candidate character strings and to assign POS to the selected strings in a ruled-based system. This mechanism avoids the sparse data problem of pure statistical approaches and the over-generation problem of rule-based approaches. It improves parser coverage and provides a tool for the lexical acquisition of new words.

1 Introduction

In this paper, new words refer to newly coined words, occasional words and other rarely used words that are neither found in the dictionary of a natural language processing system nor recognized by the derivational rules or proper name identification rules of the system. Typical examples of such words are shown in the following sentences, with the new words underlined in bold.

有人要学好的东西，也有人要**拣拾**“垃圾”。
国安队只有胡建平**冷射**得手。
中国**女垒**获企业支持。
他们还将在一些城市举办世界杯奖杯真身**巡展**。
维初越野队的健将非常努力。
警方和刑事情报部门了解**球痞**们的活动。
这对美国人的肥胖症起了**助肥**作用。
他立即去附近一家**驾校**报了名。
邵逸夫**捐建**了此楼。
柔枝低低**弯垂**去**回吻**那多情的河水。
那是一杯色彩**鲜丽**的“鸡尾酒”
中国**健力宝**青年队旗开得胜。
西藏的**晒佛节**、**传昭**等也得到恢复。
他们突然一个个变成**牛高马大**的“老美”。
女子在田径场上**争金夺银**。

The automatic identification of such words by a machine is a trivial task in languages where words are separated by spaces in written texts. In languages like Chinese, where no word boundary exists in written texts, this is by no means an easy job. In many cases the machine will not even realize that there is an unfound word in the sentence since most single Chinese characters can be words by themselves.

Purely statistical methods of word segmentation (e.g. de Marcken 1996, Sproat et al 1996, Tung and Lee 1994, Lin et al (1993), Chiang et al (1992), Lua, Huang et al, etc.) often fail to identify those words because of the sparse data problem, as the likelihood for those words to appear in the training texts is extremely low.

There are also hybrid approaches such as (Nie dt al 1995) where statistical approaches and heuristic rules are combined to identify new words. They generally perform better than purely statistical segmenters, but the new words they are able to recognize are usually proper names and other relatively frequent words. They require a reasonably big training corpus and the performance is often domain-specific depending on the training corpus used.

Many word segmenters ignore low-frequency new words and treat their component characters as independent words, since they are often of

little significance in applications where the structure of sentences is not taken into consideration. For in-depth natural language understanding where full parsing is required, however, the identification of those words is critical, because a single unidentified word can cause a whole sentence to fail.

The new word identification mechanism to be presented here is used in a wide coverage Chinese parser that does full sentence analysis. It assumes the word segmentation process described in Wu and Jiang (1998). In this model, word segmentation, including unfound word identification, is not a stand-alone process, but an integral part of sentence analysis. The segmentation component provides a word lattice of the sentence that contains all the possible words, and the final disambiguation is achieved in the parsing process.

In what follows, we will discuss two hypotheses and their implementation. The first one concerns the selection of candidate strings and the second one concerns the assignment of parts of speech (POS) to those strings.

2 Selection of candidate strings

2.1 Hypothesis

Chinese used to be a monosyllabic language, with one-to-one correspondences between syllables, characters and words, but most words in modern Chinese, especially new words, consist of two or more characters. Of the 85,135 words in our system's dictionary, 9217 of them are monosyllabic, 47778 are disyllabic, 17094 are tri-syllabic, and the rest has four or more characters. Since hardly any new character is being added to the language, the unfound words we are trying to identify are almost always multiple character words. Therefore, if we find a sequence of single characters (not subsumed by any words) after the completion of basic word segmentation, derivational morphology and proper name identification, this sequence is very likely to be a new word. This basic intuition has been discussed in many papers, such as Tung and Lee (1994). Consider the following sentence.

(1) 维初男队中锋范建毅居然冷射得手。

This sentence contains two new words (not including the name 范建毅 which is recognized by the proper name identification mechanism) that are unknown to our system:

维初 (probably the abbreviated name of a junior high school)
冷射 (a word used in sports only but not in our dictionary)

Initial lexical processing based on dictionary lookup and proper name identification produces the following segmentation:

维 初 男队 中锋 范建毅 居然 冷 射 得手

where 维初 and 冷射 are segmented into single characters. In this case, both single character-strings are the new words we want to find.

However, not every character sequence is a word in Chinese. Many such sequences are simply sequences of single-character words. Here is an example:

(2) 她回了国才会来看你。

After dictionary look up, we get

她 回 了 国 才 会 来 看 你

which is a sequence of 10 single characters. However, every character here is an independent word and there is no new word in the sentence. From this we see that, while most new words show up as a sequence of single characters, not every sequence of single characters forms a new word. The existence of a single-character string is the necessary but not sufficient condition for a new word. *Only those sequences of single characters where the characters are unlikely to be a sequence of independent words are good candidates for new words*.

2.2 Implementation

The hypothesis in the previous section can be implemented with the use of the Independent Word Probability (*IWP*), which can be a property of a single character or a string of characters.

2.1.1 Defining IWP

Most Chinese characters can be used either as independent words or component parts of multiple character words. The IWP of a single character is the likelihood for this character to appear as an independent word in texts:

$$IWP(c) = \frac{N(Word(c))}{N(c)}$$

where *N(Word(c))* is the number of occurrences of a character as an independent word in the sentences of a given text corpus and *N(c)* is the total number of occurrence of this character in the same corpus. In our implementation, we computed the probability from a parsed corpus where we went through all the leaves of the trees, counting the occurrences of each character and the occurrences of each character as an independent word.

The parsed corpus we used contains about 5,000 sentences and was of course not big enough to contain every character in the Chinese language. This did not turn out to be a major problem, though. We find that, as long as all the frequently used single-character words are in the corpus, we can get good results, for what really matters is the *IWP* of this small set of frequent characters/words. These characters/words are bound to appear in any reasonably large collection of texts.

Once we have the *IWP* of individual characters (*IWP(c)*), we can compute the *IWP* of a character string (*IWP(s)*). *IWP(s)* is the probability of a sequence of two or more characters being a sequence of independent words. This is simply the joint probability of the *IWP(c)* of the component characters.

2.1.2 Using IWP

With *IWP(c) and IWP(s)* defined , we then define a threshold *T* for *IWP*. A sequence *S* of two or more characters is considered a candidate for a new word only if its *IWP(s)* < *T* . When *IWP(s)* reaches *T*, the likelihood for the characters to be a sequence of independent words is too high and the string will not be considered to be a possible new word. In our implementation, the value of *T* is empirically determined. A lower *T* results in higher precision and lower recall while a higher *T* improves recall at the expense of precision. We tried different values and weighed recall against precision until we got the best performance. 维初 and 冷射 in Sentence (1) are identified as candidate dates because *IWP(s)*(维初) = 8% and *IWP(s)*(冷射) = 10% while the threshold is 15%. In our system, precision is not a big concern at this stage because the final filtering is done in the parsing process. We put recall first to ensure that the parser will have every word it needs. We also tried to increase precision, but not at the expense of recall.

3 POS Assignment

Once a character string is identified to be a candidate for new word, we must decide what syntactic category or POS to assign to this possible new word. This is required for sentence analysis where every word in the sentence must have at least one POS.

3.1. Hypothesis

Most multiple character words in Chinese have word-internal syntactic structures, which is roughly the POS sequence of the component characters (assuming each character has a POS or potential POS). A two-character verb, for example, can have a V-V, V-N, V-N or A(dv)-V internal structure. For a two-character string to be assigned the POS of verb, the POS/potential POS of its component characters must match one of those patterns. However, this matching alone is not the sufficient condition for POS assignment. Considering the fact that a single character can have more than one POS and a single POS sequence can correspond to the internal word structures of different parts of speech (V-N can be verb or a noun, for instance), simply assigning POS on the basis of word internal structure will result in massive over-generation and introduce too much noise into the parsing process. To prune away the unwanted guesses, we need more help from statistics.

When we examine the word formation process in Chinese, we find that new words are often modeled on existing words. Take the newly coined verb 冷射 as an example. Scanning our dictionary, we find that 冷 appears many times as the first character of a two-character verb, such as 冷冻, 冷凝, 冷笑, 冷藏, 冷轧, 冷敷, etc. Meanwhile, 射 appears many times as the second

character of a two-character verb, such as 俯射, 反射, 平射, 折射, 斜射, 直射, etc. This leads us to the following hypothesis:

A candidate character string for a new word is likely to have a given POS if the component characters of this string have appeared in the corresponding positions of many existing words with this POS.

3.2. Implementation

To represent the likelihood for a character to appear in a given position of a word with a given POS and a given length, we assign probabilities of the following form to each character:

P*(Cat,Pos,Len)*

where Cat is the category/POS of a word, Pos is the position of the character in the word, and Len is the length (number of characters) of the word. The probability of a character appearing as the second character in a four-character verb, for instance, is represented as ***P****(Verb,2,4)*.

3.1.1. Computing P(Cat,Pos,Len)

There are many instantiations of ***P****(Cat,Pos,Len)*, depending on the values of the three variables. In our implementation, we limited the values of *Cat* to Noun, Verb and Adjective, since they are the main open class categories and therefore the POSes of most new words. We also assume that most new words will have between 2 to 4 characters, thereby limiting the values of *Pos* to 1-4 and the values of *Len* to 2-4. Consequently each character will have 27 different kinds of probability values associated with it. We assign to each of them a 4-character name where the first character is always "P", the second the value of *Cat*, the third the value of *Pos*, and the fourth the value of *Len*. Here are some examples:

Pn12 (the probability of appearing as the first character of a two-character noun)

Pv22 (the probability of appearing as the second character of a two-character verb)

Pa34 (the probability of appearing as the third character of a four-character adjective)

The values of those 27 kinds of probabilities are obtained by processing the 85,135 headwords in our dictionary. For each character in Chinese, we count the number of occurrences of this character in a given position of words with a given length and given category and then divide it by the total number of occurrences of this character in the headwords of the dictionary. For example,

$$Pv12(c) = \frac{N(v12(c))}{N(c)}$$

where *N(v12(c))* is the number of occurrences of a character in the first position of a two-character verb while *N(c)* is the total number of occurrences of this character in the dictionary headwords. Here are some of the values we get for the character射:

Pn12(射) = 7%	*Pn22*(射) = 0%
Pv12(射) = 3%	*Pv22*(射) =24%
Pv23(射) = 39%	*Pa22*(射) =1%

It is clear from those numbers that the character 射 tend to occur in the second position of two-character and three-character verbs.

3.1.2. Using P(Cat,Pos,Len)

Once a character string is identified as a new word candidate, we will calculate the POS probabilities for the string. For each string, we will get *P(noun)*, *P(verb)* and *P(adj)* which are respectively the probabilities of this string being a noun, a verb or an adjective. They are the joint probabilities of the *P(Cat,Pos,Len)* of the component characters of this string. We then measure the outcome against a threshold. For a new word string to be assigned the syntactic category *Cat*, its *P(Cat)* must reach the threshold. The threshold for each *P(Cat)* is independently determined so that we do not favor a certain POS (e.g. Noun) simply because there are more nouns in the dictionary.

If a character string reaches the threshold of more than one *P(Cat)*, it will be assigned more than one syntactic category. A string that has both *P(noun)* and *P(verb)* reaching the threshold, for example, will have both a noun and a verb added to the word lattice. The ambiguity is then resolved in the parsing process. If a string passes the *IWP* test but fails the *P(Cat)* test, it will

receive noun as its syntactic category. In other words, the default POS for a new word candidate is noun. This is what happened to 维初 in the Sentence (1). 维初 passed the *IWP* test, but failed each of the *P(Cat)* tests. As a result, it is made a noun by default. As we can see, this assignment is the correct one (at least in this particular sentence).

4. Results and Discussion

4.1. Increase in Parser Coverage

The new word identification mechanism discussed above has been part of our system for about 10 months. To find out how much contribution it makes to our parser coverage, we took 176,863 sentences that had been parsed successfully with the new word mechanism turned on and parsed them again with the new word mechanism turned off. When we did this test at the beginning of these 10 months, 37640 of those sentences failed to get a parse when the mechanism was turned off. In other words, 21.3% of the sentences were "saved" by this mechanism. At the end of the 10 months, however, only 7749 of those sentences failed because of the removal of the mechanism. At first sight, this seems to indicate that the new word mechanism is doing a much less satisfactory job than before. What actually happened is that many of the words that were identified by the mechanism 10 months ago, especially those that occur frequently, have been added to our dictionary. In the past 10 months, we have been using this mechanism both as a component of robust parsing and as a method of lexical acquisition whereby new entries are discovered from text corpora. This discovery procedure has helped us find many words that are found in none of the existing word lists we have access to.

4.2. Precision of Identification

Apart from its contribution to parser coverage, we can also evaluate the new word identification mechanism by looking at its precision. In our evaluation, we measured precision in two different ways.

In the first measurement, we compared the number of new words that are proposed by the guessing mechanism and the number of words that end up in successful parses. If we use *NWA* to stand for the number of new words that are added to the word lattice and *NWU* for the number of new words that appear in a parse tree, the precision rate will be *NWU / NWA*. Actual testing shows that this rate is about 56%. This means that the word guessing mechanism has over-guessed and added about twice as many words as we need. This is not a real problem in our system, however, because the final decision is made in the parsing process. The lexical component is only responsible for providing a word lattice of which one of the paths is correct. In the second measurement, we had a native speaker of Chinese go over all the new words that end up in successful parses and see how many of them sound like real words to her. This is a fairly subjective test but nonetheless meaningful one. It turns out that about 85% of the new words that "survived" the parsing process are real words.

We would also like to run a large-scale recall test on the mechanism, but found it to be impossible. To run such a test, we have to know how many unlisted new words actually exist in a corpus of texts. Since there is no automatic way of knowing it, we would have to let a human manually check the texts. This is too expensive to be feasible.

4.3. Contributions of Other Components

While the results shown above do give us some idea about how much contribution the new word identification mechanism makes to our system, it is actually very difficult to say precisely how much credit goes to this mechanism and how much to other components of the system. As we can see, the performance of this mechanism also depends on the following two factors:

(1) The word segmentation processes prior to the application of this mechanism. They include dictionary lookup, derivational morphology, proper name identification and the assembly of other items such as time, dates, monetary units, address, phone numbers, etc. These processes also group characters into words. Any improvement in those components will also improve the performance of the new word mechanism. If every word that "should" be found by

those processes has already been identified, the single-character sequences that remain after those processes will have a better chance of being real words.

(2) The parsing process that follows. As mentioned earlier, the lexical component of our system does not make a final decision on "wordhood". It provides a word lattice from which the syntactic parser is supposed to pick the correct path. In the case of new word identification, the word lattice will contain both the new words that are identified and the all the words/characters that are subsumed by the new words. A new word proposed in the word lattice will receive its official wordhood only when it becomes part of a successful parse. To recognize a new word correctly, the parser has to be smart enough to accept the good guesses and reject the bad guesses. This ability of the parser will imporve as the parser improves in general and a better parser will yield better final results in new word identification.

Generally speaking, the mechanisms using *IWP* and *P(Cat,Pos,Len)* provide the *internal* criteria for wordhood while word segmentation and parsing provide the *external* criteria. The internal criteria are statistically based whereas the external criteria are rule-based. Neither can do a good job on its own without the other. The approach we take here is not to be considered statistical natural language processing, but it does show that a rule-based system can be enhanced by some statistics. The statistics we need can be extracted from a very small corpus and a dictionary and they are not domain dependent. We have benefited from the mechanism in the analysis of many different kinds of texts.

References

Chang, Jyun-Sheng, Shun-Der Chen, Sue-Jin Ker, Ying Chen and John S. Liu (1994) A multiple-corpus approach to recognition of proper names in Chinese texts, *Computer Processing of Chinese and Oriental Languages*, Vol. 8, No. 1 pp. 75-85.

Chen, Keh-Jiann and Shing-Huan Liu (1992). Word identification for Mandarin Chinese sentences, Proceedings of COLING-92, pp. 23-28.

Chiang, T. H., Y. C. Lin and K.Y. Su (1992). Statisitical models for word segmentation and unknown word resolution, *Proceedings of the 1992 R. O. C. Computational Linguistics Conference*, 121-146, Taiwan.

De Marcken, Carl (1996). *Unsupervised Language Acquisition*, Ph.D dissertation, MIT.

Lin, M. Y. , T. H. Chiang and K. Y. Su (1993) A prelimnary study on unknown word problem in Chinese word segmentation, *Proceedings of the 1993 R. O. C. Computational Linguistics Conference*, 119-137, Taiwan.

Lua, K T. Experiments on the use of bigram mutual information in Chinese natural language processing.

Nie, Jian Yun, et al. (1995) Unknown Word Detection and Segmentation of Chinese using Statistical and Heuristic Knowledge, Communications of COLIPS, vol 5, No. 1 &2, pp.47, Singapore.

Sproat, Richard, Chilin Shih, William Gale and Nancy Chang (1996). A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, Volume 22, Number 3.

Tung, Cheng-Huang and Lee His-Jian (1994). Identification of unknown words from a corpus. *Computer Processing of Chinese and Oriental Languages*, Vol. 8 Supplement, pp. 131-145.

Wu, Andi and Zixin Jiang (1998) Word segmentation in sentence analysis, *Proceedings of the 1998 International Conference on Chinese Information Processing*, pp. 169-180.

Yeh, Ching-Long and His-Jian Lee (1991). Rule-based word identification for Mandarin Chinese sentences – a unification approach, *Computer Processing of Chinese and Oriental Languages*, Vol 5, No 2, Page 97-118.