Design of CKIP Chinese Word Segmentation System

Wei-Yun Ma, Keh-Jiann Chen

Institute of Information Science

Academia Sinica, Taipei

weiyun.ma@msa.hinet.net, kchen@iis.sinica.edu.tw

ABSTRACT

In this paper, we describe the design of the CKIP Chinese word segmentation system and analyse its performance. The system utilizes a modulized approach. Independent modules were designed to solve the problems of segmentation ambiguities and identifying unknown words. Segmentation ambiguities are resolved by a hybrid method of using heuristic and statistical rules. Regular-type unknown words are identified by regular expressions and irregular types of unknown words are detected first by their occurrence and then extracted by morphological rules with statistical and morphological constraints. At the first international Chinese Word Segmentation Bakeoff, the CKIP system was tested on open and closed tracks of Beijing University (PK) and Hong Kong CityU (HK). The evaluation results show our system performed very well on both the HK open track and closed tracks; and was acceptable on the PK tracks.

1. Introduction

At the first international Chinese Word Segmentation Bakeoff, Academia Sinica participated in testing on open and closed tracks of Beijing University (PK) and Hong Kong CityU (HK). The same segmentation algorithm was applied to process these two corpora, except for character code conversion from GB to BIG5 for the PK corpus. Also, a few modifications were made due to different segmentation standards. The difference between open and closed tracks is that while processing the closed track, segmentation algorithms and lexicons have to be trained by the provided corpora only.

It is well known that there are two major difficulties in Chinese word segmentation. One is resolving the ambiguous segmentation, and the other is identifying unknown words. Our earlier work focused primarily on resolving segmentation ambiguities and using regular expressions to handle the determinant-measure and reduplication compounds (Chen & Liu 1992, Chen 1999). We adopted a variation of the longest matching algorithm with several heuristic rules to resolve the ambiguities and achieve a 99.77% success rate, without counting the mistakes that occurred due to the existence of unknown words. After that, we paid more attention to the problems of extracting and identifying unknown words (Chen et.al 1997, Chen & Bai 1998, Chen & Ma 2002, Tseng & Chen 2002, Ma & Chen 2003). The process of unknown word extraction can be roughly divided into two steps, i.e. the detection process and the extraction process. The detection process detects possible occurrences of unknown words (Chen & Bai 1998), so that deeper morphological analysis can only be carried out at the places where unknown word morphemes are detected (Chen & Ma 2002, Ma & Chen 2003).

In the following sections, in addition to the bakeoff results evaluated by SIGHAN, we also present some other relevant experiment results and provide performance analysis of the system

2. System Overview

Modulized approaches were adopted for the CKIP word segmentation system. Independent modules were designed and to solve the problems of word matching, regular-type compound word generation, segmentation ambiguities, unknown word detection and unknown word identification. Figure 1 illustrates the block diagram. The first two steps of the word segmentation algorithm are word matching and resolution for ambiguous matches. These two processes are performed in parallel. The algorithm reads the input sentences from left to right and matches the input character string with lexical words and compounds generated by regular expressions, such as numbers, determinative-measure compounds and reduplications. If an ambiguous segmentation does occur, the matching algorithm looks ahead two more words and the disambiguation rules for three word chunks are then applied (Chen & Liu 1992).

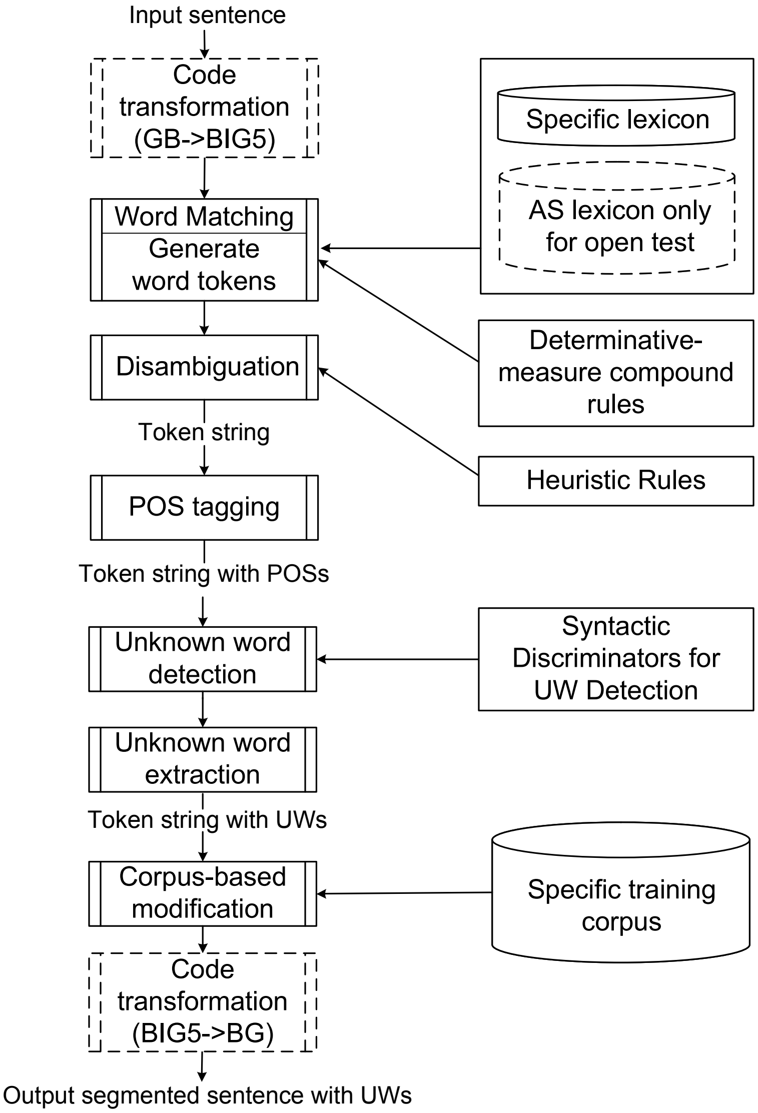

Figure 1. Flowchart of the System

For instance, in (1), the first matched word could be '完' or '完成'. Then, the algorithm will look ahead and consider all of the possible combinations of three word chunks, as shown in (2).

(1) 完成 鑑定 報告
complete authenticate report
"complete the report about authenticating"

(2) 完 成 鑑定
完成 鑑定 報
完成 鑑定 報告

The disambiguation algorithm will select the first word of the most plausible chunks as the solution, according to heuristic rules. Details of heuristic rules will be presented in Section 3. In this case, the most plausible word sequence is “完成 鑑定 報告”. So, its first word “完成” is selected. The algorithm then proceeds to process the next word until all the input text has been processed. After the disambiguation process, an input sentence is segmented into a word sequence. Then, for the purpose of unknown word detection and extraction, a Pos bi-gram tagging model is applied to tag Pos of words.

It is clear that unknown words in the input text will be segmented into several adjacent tokens (known words, or monosyllabic morphemes). At the unknown word detection stage, every monosyllable is processed and determined as either a word or an unknown word morpheme by a set of syntactic discriminators, which are trained from a word segmented corpus.

(3) 若 能 提升 毛利率
if can increase gross profit rate
"if gross profit rate can be increased…"

(4) after first step word segmentation:
若 能 提升 毛 利 率
after unknown word detection:
若 能 提升 毛(?) 利(?) 率(?)
after unknown word extraction:
若 能 提升 毛利率

For example, the correct segmentation of (3) is shown, but the unknown word ”毛利率” is segmented into three monosyllabic words after the first step of the word segmentation process. In (4), unknown word detection processes will mark the sentence as “若() 能() 提升() 毛(?) 利(?) 率(?)”, where (?) denotes the detected monosyllabic unknown word morpheme and () denotes the common known word. Details of the detection process will be introduced in Section 4. At next stage of the extracting process, the rule matching process only focuses on the morphemes marked with (?) and tries to combine them with left/right neighbors according to the morphological rules for unknown words. The unknown word “毛利率” is then extracted. The extraction technology will be addressed in Section 5.

3. Segmentation Models Based on Heuristic Rules

As mentioned earlier, the algorithm reads the input sentences from left to right and matches the input character string with lexemes and regular expressions for compound words. If an ambiguous segmentation does occur, the matching algorithm looks ahead two more words and utilizes the heuristic disambiguation rules to select the first word of the most plausible chunks as the solution. It then proceeds to process the next word until all the input text has been processed.

The most powerful and commonly used disambiguation rule is the heuristic rule of longest matching. There are a few variations of longest matching rules; the following longest matching rule was adopted by Chen & Liu, 1992.

Rule 1 - Longest Matching Rule: The most plausible segmentation is the three word sequence with the longest length.

This is equivalent to finding the chunk with the maximum value of Length(W1)+ Length(W2)+Length(W3), where W1, W2, and W3 are three words in a chunk. In the above example(2), the longest matched three-word chunk is the third sequence, “完成 鑑定 報告”. Therefore, the first segmented word, “完成”, is identified. This heuristic rule achieves above 99% accuracy and has a high applicability of 93% in evaluating the Academia Sinica corpus. This means that 93% of the ambiguities were resolved by this rule. However, there are still about 7% of ambiguities, i.e. the three word chunks with the same length, but with different segmentations, which cannot be resolved by the maximal matching rule. The following heuristic rules were used for further resolution.

Rule 2 - Word Length Rule: Picks the three-word chunk that has the smallest standard deviation in the length of the three words.

This is equivalent to finding the chunk with the minimal value of (L(W1)-Mean)**2 + (L(W2)-Mean)**2 + (L(W3)-Mean)**2, where W1, W2, and W3 are three words in a chunk. “Mean” is the average length of the word W. The word length rule simply states that the word length is usually evenly distributed. For instance in (5), the segmentation of (5a) has the value 0, but (5b) has value 2. Therefore, according to the word length rule, (5a) will be the selected solution and, indeed, it is the correct segmentation.

(5) 研究 生命 起源

research life origin

"to investigate the origin of life"

a. 研究 生命 起源

b. 研究生 命 起源

However it may happen that there are more than two chunks with the same length and variance, so we need a further resolution.

Rule 3 - Morphemic Rules:

(i). Pick the chunk with fewer bound morphemes.

(ii). Pick the chunk with fewer determinative-measure compounds (DM).

That is to say, normal words get higher priority than bound morphemes and DMs. For example, (6) and (7) were resolved by rule (i) and rule (ii) respectively. Since “續” in (6b) is a bound morpheme and “他本” is a DM in (7b) and neither any bound morpheme nor any DM occurs in (6a) and (7a), (6a) and (7a) will be selected as the right choices. (In this case, they are.)

(6) 協調 上 手續 較 麻煩
negotiate up procedure more trouble
"In negotiation, the process is more complicated.”
a. 協調 上 手續 較 麻煩
b. 協調 上手 續 較 麻煩

(7) 他 本人
he self
“he himself”
a. 他 本人
b. 他本 人

Rule 4 - Probability Rule:

(a). Pick the chunk with the highest frequency of monosyllabic words.
(b). Pick the chunk with the highest probability value.

The probability rule is similar to the statistical word segmentation method. The only difference is that the probability rule is applied locally on a tree-word chunk, but the scope of the general statistical method is a whole sentence.

The above mentioned heuristic rules achieve very high accuracy. Each rule was designed to solve different kinds of ambiguities. The longest matching rule is the kernel of the heuristic method and has to be applied first. The other rules play complementary roles to resolve about 7% of the remaining ambiguities, after application of the maximal matching rule. Chen & Liu (1992) reported an accuracy rate of 99.66% of the system’s total performance, without considering the occurrence of unknown words.

4. Unknown Word Detection

For better focusing, while extracting unknown words, morphological rules or statistical rules are applied only in places where unknown words are detected. An unknown word detection method proposed by Chen & Bai (1998) is applied in our system. In most cases, after the dictionary look-up segmentation process, each unknown word is segmented into a sequence of shorter words or morphemes, which contain at least one monosyllabic morpheme. Therefore, the occurrence of monosyllabic morphemes (i.e. single character words) in a segmented input text may denote the possible existence of unknown words. However, if all occurrences of monosyllabic words are considered as morphemes of unknown words, the recall rate of the detection will be about 99%, but the precision rate could be as low as 13.4%. Consequently, a method to distinguish between monosyllabic words and monosyllabic morphemes (i.e. part of unknown words) is required. Our system detects monosyllabic known-words, instead of monosyllabic morphemes, since we can check the syntactic validity of monosyllabic known-words.

The adopted unknown word detection method (Chen & Bai 1998) is a corpus-based learning algorithm, which derives a set of syntactic discriminators. The syntactic discriminators are used to distinguish whether a monosyllable is a word, or an unknown word morpheme. Chen and Bai (1998) adopted ten types of context rule patterns, shown in Table 1, to generate discriminator rule instances from a training corpus. The rule instances are used to check the syntactic validity of a word. The training corpus also provides the applicability and accuracy of each rule instance. Each rule contains a key token within curly brackets and its contextual tokens without brackets. For some rules, there may be no contextual dependencies. The function of each rule means that in a sentence, if a character and its context match the key token and the contextual tokens of the rule respectively, this character is a common word (i.e. not a morpheme of an unknown word). For instance, the rule "{Da} VHC" says that a character with syntactic category[1] Da is a common word, if it is followed by a word of syntactic category VHC.

Rule type	Example rule instance
{char}	{的}
word {char}	不 {願}
{char} word	{全} 世界
{category}	{T}
{category} category	{Da} VHC
category {category}	Na {VCL}
{char} category	{就} VH
category {char}	Na {上}
category category {char}	Nh P {書}
{char} category category	{極} VH T

Table1. Rule Types and Rule Instances

The corpus-based learning approach has the advantages of: 1. automatic rule learning, 2. automatic evaluation of the performance of each rule, and 3. balancing of recall and precision rates through dynamic rule set selection. There are tradeoffs between precision and recall for selecting different rule sets. To serve the purpose of unknown word detection, we prefer rule sets with a higher recall rate. Therefore, a rule set of with a detection rate of 96% and a precision rate of 60% was adopted. Where the detection rate is 96%, it means that for 96% of unknown words, at least one of their morphemes is detected as part of an unknown word. The precision of 60% means that 60% of detected morphemes are genuine unknown word morphemes. Although the precision is not high, most instances of over-detecting errors are "isolated", which means there are few situations when two adjacent detected monosyllabic unknown morphemes are both wrong at the same time. This operative characteristic, which is very important in the design of general morphologic rules for unknown words, is described in the section 5.2.1

5. Unknown Word Extraction

At the detection stage, the contextual rules are applied to detect the occurrence of unknown words. Hence, the extraction process can be more focused. At the extraction

[1] The syntactic category symbols here are based on CKIP, 1993. The meaning of each category we have adopted is as follows: T(particle), Da(adverb of quantity), VHC(stative motion verb), Na(noun), VCL(active transitive verb with locative object), VH(stative intransitive verb), Nh(pronoun), P(preposition) and so on.

stage, the extraction rules will be triggered by detected morphemes only. To avoid over-generation, the design of the extraction rules not only targets a high recall rate, but also tries to maintain high precision at the same time. Therefore, unknown word extraction rules will be context, content, and statistically constrained. In this system, morphological rules for certain specific types of unknown words were designed, including the rules for Chinese personal names, foreign transliteration names and compound nouns (See Section 5.1). In order to increase the coverage of extraction rules, we have developed a set of general morphological rules to identify all kinds of unknown words, without differentiating their types. The design and application of these general morphological rules will be addressed in Section 5.2.

Since the precision of the specific-type morphological rules is better than the precision of the general morphological rules, we first try to apply specific-type morphological rules to extract unknown words with certain types and then apply general morphological rules to recover unidentified unknown words.

5.1 Specific Morphological Rules

We designed specific rules for three different types of unknown words, namely: Chinese personal names, foreign transliteration names, and compound nouns with common affixes. We will not go into the detailed extraction process for each different type. It will be exemplified by the Chinese personal name extraction to illustrate the idea of using different clues in the extraction process. First of all, when the content information is used, each different type of unknown word has its own morphological structure. For instance, a typical Chinese personal name starts with a last name, followed by a given name. There are about one hundred last names. Most of them have common characters. Given names are usually one or two characters and seldom with bad implication. Based on the above structured information about Chinese personal names, the following name extraction rules were designed: (?) denotes the detected monosyllabic unknown word morpheme and () denotes the common known word (See Table2). Context information is used to verify and determine the boundary of the extracted word. For example, in the last rule of Table 2, context information and statistical information are used to resolve the ambiguity of the word boundary. This is illustrated by the following example.

1) after detection : 張(?) 明(?) 正() 要() 殺() 人()。
extractnion : 張明正 要 殺 人。
Ming-Zheng Zhang want kill somebody.
or 張明 正 要 殺 人。
Ming Zhang just want kill somebody.

In this example, there are two possible candidates for personal names, “張明” and “張明正”. According to the context information, the bi-gram of (NAME,正), i.e. a personal name followed by a word 正, is less frequent than the bi-gram of (NAME, 要) in the corpus. So without considering statistical constraints, it would suggest that “張明正”, instead of “張明”, is the correct extraction. The locality of the keywords is also a very important clue for identification, since the keywords of a text are usually unknown words and frequently recur in the text. This characteristic is utilized to resolve extraction ambiguities. For instance, if an another sentence “張(?) 明(?) 來() 了()” occurs in the same text, it suggests “張明” is the correct extraction, since the statistical constraint $prob_{document}(正\,|\,張明) < 1$ rejects “張明正”.

Rule type	Constraints & Procedure
$ms_i(?)\ ms_{i+1}(?)\ ms_{i+2}(?)$	$combine(i, i+1, i+2)$
$ms_i()\ ms_{i+1}(?)\ ms_{i+2}(?)$	$combine(i, i+1, i+2)$
$ms_i(?)\ ms_{i+1}()\ ms_{i+2}(?)$	$combine(i, i+1, i+2)$
$ds_i()\ ms_{i+1}(?)$	$combine(i, i+1)$
$ms_i(?)\ ds_{i+1}()$	$combine(i, i+1)$
$ms_i(?)\ ms_{i+1}(?)\ ps_{i+2}()$	$combine(i, i+1)$
$ms_i(?)\ ms_{i+1}(?)\ ms_{i+2}()$	as follows:

```
if $prob_{document}(ms_{i+2} \mid ms_i ms_{i+1}) < 1$
    $combine(i, i+1)$ as a disyllabic name
elsif $freq_{coupus}(NAME, ms_{i+2}, word_{i+3}) \geq 1$
        $combine(i, i+1)$ as a disyllabic name
    elsif $freq_{coupus}(NAME, word_{i+3}) \geq freq_{coupus}(NAME, ms_{i+2})$
            $combine(i, i+1, i+3)$ as a trisyllabic name
        else $combine(i, i+1)$ as a disyllabic name
```

Notes: *ms(?)* denotes a detected monosyllabic unknown word morpheme; *ms()* denotes a monosyllabic common word; *ds()* denotes a disyllabic known word; *ps()* denotes a polysyllabic known word, which consists of more than one syllable; *word* denotes the known word, which could consist of any number of syllables and ms_i must belong to the Common Chinese Last Name Set, such as 陳, 王…etc.

Table 2. Rule Types for Chinese Personal Names

5.2 General Morphological Rules

Although specific morphological rules work well in regular-type unknown word extractions, it's difficult to develop morphological rules for irregular unknown words. In this section, we present a common structure for unknown words from another point of view. An unknown word is regarded as the combination of morphemes, which are consecutive morphemes/words in context after segmentation, most of which are monosyllables. We adopt context free grammar (Chomsky 1956), which is the most commonly used generative grammar for modelling constituent structures, to express our unknown word structure.

5.2.1 Rule Derivation

As discussed in Section 4, for 96% of unknown words, at least one of morpheme is detected as part of an unknown word. We, therefore, represent unknown word structures with the constraint that they contain at least one detected morpheme. Taking this constraint into consideration, the rules for modeling unknown words are shown in the following example of an unknown word representation.

==

```
UW    →   UW UW           (1)
        | ms(?) ms(?)     (2)
        | ms(?) ps()      (3)
        | ms(?) ms()      (4)
        | ps() ms(?)      (5)
        | ms() ms(?)      (6)
        | ms(?) UW        (7)
        | ms() UW         (8)
        | ps() UW         (9)
        | UW ms(?)        (10)
        | UW ms()         (11)
        | UW ps()         (12)
```

Notes: There is one non-terminal symbol, “UW”, which denotes “unknown word” and is also the start symbol. There are three terminal symbols: namely: ms(?), which denotes the detected monosyllabic unknown word morpheme; ms(), which denotes the monosyllabic common known word; and ps(), which denotes the polysyllabic (more than one syllable) known word.

==

Table 3. General Morphologic Rules for Unknown Words

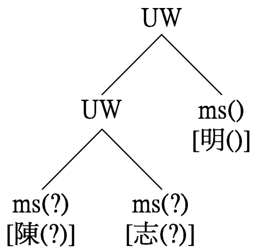

Figure 2. A possible structure for the unknown word “陳志明”(Chen Zhi Ming), which is segmented initially and detected as “陳(?) 志(?) 明()”, and “明” was marked incorrectly at the detection stage. This structural representation utilizes rule (2) and rule (11).

There are three kinds of commonly used measures applied to evaluate grammar: 1. Generality (recall): The range of sentences the grammar analyzes correctly. 2. Selectivity (precision): The range of non-sentences it identifies as problematic. 3. Understandability: the simplicity of the grammar itself (Allen 1995). For generality, 96% of unknown words have the structure matched by the general morphologic rules, so the grammar has high generality to generate unknown words. But for selectivity, our rules are over-generation. Many patterns accepted by the rules are not words. The main reason is that the rules have to include non-detected morphemes for high generality. Therefore, selectivity is sacrificed momentarily. In the next section, rules would be further constrained by linguistic and text-based statistical constraints to compensate for the selectivity of the grammar. For understandability, each rule in (1)-(12) consists of just two right-hand side symbols. The reason for using this kind of presentation is that it regards the unknown word structure as a series of combinations of two consecutive morphemes. We can, therefore, simplify the analysis of an unknown word structure by only analyzing its combinations of two consecutive morphemes.

5.2.2 Constraints

Since the general morphologic rules in Table 3 have a high generality and a low selectivity to model unknown words, we append some constraints to restrict their application. There are tradeoffs between generality and selectivity: higher selectivity usually results in lower generality. In order to maintain high generality while assigning constraints, we assign different constraints to different rules according to their characteristics, such that generality is only degraded slightly but selectivity is upgraded significantly.

The rules in Table 3 are classified as follows: A) Rules in which both right-hand side symbols consist of detected morphemes, i.e, (1), (2), (7), and (10). B) Rules in which just one of the right-hand side symbols consists of detected morphemes, i.e, (3), (4), (5), (6), (8), (9), (11), and (12). The former is regarded as a “strong” structure, since it is considered to have more possibility of composing an unknown word, or an unknown word morpheme. The latter is regarded as a “weak” structure, because it is considered to have less possibility of composing an unknown word, or an unknown word morpheme. The basic idea is to place more constraints on those rules with a weak structure and less constraints on those rules with a strong structure.

The constraints we applied included word length, linguistic and statistical constraints. For statistical constraints, since our strategy is to utilize the locality of unknown words in a text, we use text-based statistical measures as statistical constraints. It is well known that keywords often reoccur in a document (Church 2000) and very often the keywords are also unknown words. Therefore, the reoccurrence frequency within a text is adopted as the constraint. Another useful statistical phenomenon in a text is that a polysyllabic morpheme is very unlikely to be the morphemes of two different unknown words within the same text. Hence, we restrict the rule with polysyllabic symbols by evaluating the conditional probability of those symbols. In addition, syntactic constraints are also utilized here. The syntactic categories of most unknown word morphemes, belong to “bound”, “verb”, ”noun”, and “adjective”, instead of “conjunction”, “preposition”…etc. So we restrict the rule with non-detected symbols by checking whether the syntactic categories of its non-detected symbols belong to “bound”, “verb”, ”noun”, or “adjective”. To avoid unlimited recursive rule application, the length of a matched unknown word is restricted, unless there is a very strong statistical association between two matched tokens. The constraints we adopted are presented in Table 4. A rule might be restricted by one constraint, or multi-constraints, or it may not have any constraints.

$\text{Freq}_{\text{docu}}$(LR)>=Threshold	(3) (4) (5) (6) (8) (9) (11) (12)
P_{docu}(L|R)=1	(1) (3) (7) (8) (9) (12)
P_{docu}(R|L)=1	(1) (5) (9) (10) (11) (12)
Category(L) is bound, verb, noun or adjective	(5) (6) (8) (9)
Category(R) is bound, verb, noun or adjective	(3) (4) (11) (12)

Notes: L denotes left terminal of right-hand side
R denotes right terminal of right-hand side
Threshold value is dependent on the Length(LR) and text size. The basic

idea is that larger length(LR) or text size requires larger Threshold value.

Table 4. Constraints for General Morphologic Rules

5.2.3 Bottom-up Merging Algorithm

We adopted a greedy strategy to design an efficient bottom-up merging algorithm, consulting the general morphologic rules to extract unknown words. The basic idea is that for a segmented sentence, if there are many rule-matched token pairs, which also satisfy the rule constraints, the token pair with the highest co-occurrence in the text is merged first and forms a new token string. The same procedure is then applied to the updated token string recursively until no token pair satisfies the general morphologic rules. This is illustrated by the following example.

==

Text environment:

“陳志強” (Chen Zhi Qiang), an unknown word, occurs three times.
“造勢” (take an electing activity), an unknown word, occurs two times.
“陳志強造勢” (Chen Zhi Qiang took an electing activity), a sentence, occurs one time.

Input: 陳志強造勢

After initial segmentation and detection:

陳(?) 志(?) 強(?) 造(?) 勢(?)

$\boxed{3}$ 3 1 2 co-occurrence

After first iteration:

陳志(uw) 強(?) 造(?) 勢(?)

$\boxed{3}$ 1 2 co-occurrence

After second iteration:

陳志強(uw) 造(?) 勢(?)

$\boxed{2}$ co-occurrence

After third iteration:

陳志強(uw) 造勢(uw)

==

Figure 3. The Extraction Process of Input “陳志強造勢”.

By applying the general morphologic rules and the greedy strategy, as well as overlapping character pair ambiguity, the algorithm is able to deal with more complex overlapping and coverage ambiguity, even that which results from consecutive unknown words. In Finger 3, the input sentence “陳志強造勢” is segmented into two unknown words “((陳志)強)” and “(造勢)” by applying rules (2), (10) and (2) in turn. “陳志強” and ”造勢” can not be merged, since P(造勢|陳志強)<1 violates the constraint of rule (1). This applies to “陳志強” and ”造”,which do not satisfy rule (10) in the third iteration. With this simple algorithm, unknown words of any length can be extracted. During the extraction process, the boundaries of unknown words are extended iteratively, until no rule can be applied.

In our final system, we adopted specific-type morphological rules to extract regular unknown words and general rules to extract the remaining irregular unknown words. The overall performance was a 57% recall rate and a 76% precision rate. By only using the specific type of morphological rules for Chinese personal names, foreign transliteration names, and compound nouns with common affixes, a recall rate of 25% and a precision rate of 80% were achieved. The general rules improved the recall rate by 32%, without sacrificing too much precision.

6. Adaptation to Different Tracks

It is known that different segmentation standards can affect the performance of segmentation significantly. At the SIGHAN Bakeoff, due to the limited preparation time, we focused primarily on adjusting the regular expressions for determinant-measure compounds according to the HK and PK segmentation standards.

To cope with the problem of character coding difference, while processing the PK track, a shortcut method of converting GB codes to BIG5 codes was adopted. Instead of re-designing, or re-implementing the GB segmentation system, we converted the codes of training and testing PK corpora into BIG5 versions and performed the segmentation in the BIG5 environment. The segmented results were then translated back to GB codes as the final outputs. By comparison the processing of HK corpus was easier for us, because our system was designed for the BIG5 environment.

With regard to the lexicons, for the closed test, both PK and HK lexicons were derived from the word sets of each respective training corpus. For the open test, each lexicon was enhanced by adding the lexical entries in the CKIP lexicon. The sizes of the lexicons are shown in Table 5.

	HK	PK
# of lexical entries (HK/PK)for closed test	22K	50K
# of lexical entries (HK/PK join CKIP) for open test	140K	156K

Notes: # lexicon of (CKIP) is 133K

Table 5. The Sizes of Lexicons

Syntactic categories of words were utilized in the unknown word detection and extraction processes. We don't have syntactic categories for words which are not in the CKIP lexicon. Therefore, we (Chen et.al 1997, Tseng & Chen 2002) use the association strength between morphemes and syntactic categories to predict the category of a new word. The accuracy rate is about 80%.

7. Evaluation Results

There are several evaluation indices provided by SIGHAN, i.e. test recall (R), test precision (P), F score[2], the out-of-vocabulary (OOV) rate for the test corpus, the recall on OOV words (R_{oov}) and the recall on in-vocabulary (R_{iv}) words. Table 6 shows the evaluation results of our system in the HK closed and open tracks. For both tracks, our system ranked first on F scores.

	R	P	F	OOV	R_{oov}	R_{iv}
Closed	0.947	0.934	0.940	0.071	0.625	0.972
Open	0.958	0.954	0.956	0.071	0.788	0.971

Table 6. Scores for HK

The evaluation of our system on the PK closed and open tracks is shown in Table 7. For the PK closed track, our system ranked 6th among 10 systems, and for the PK open track, our system ranked 3rd among 8 systems.

	R	P	F	OOV	R_{oov}	R_{iv}
Closed	0.939	0.934	0.936	0.069	0.642	0.961
Open	0.939	0.938	0.938	0.069	0.675	0.959

Table 7. Scores for PK

Because the Academia Sinica corpus of the AS track was provided by us, we were not allowed to participate on any AS track in this contest. Nevertheless, in Table 8 we have shown the performance of our system for evaluating the AS open track. Our system ranked first when it was compared with the other participants of the AS open track.

R	P	F	OOV	R_{oov}	R_{iv}
0.968	0.966	0.967	0.022	0.657	0.975

Table 8. Scores for AS Open Track

8. Discussions and Conclusions

The evaluation results show that our system performs very well in either the HK closed track or the HK open track. We think the key to the success of our system is that our unknown word extraction performs better than other participants. This can be seen by the results of HK closed track. The 2th and 4th systems, which have better performance in R_{iv}, but worse $R_{oov,}$ than our system, perform worse than our system in f score. Furthermore, to have better performance, a high precision for unknown word extraction is necessary, since one identification error may cause at least two segmentation errors.

The importance of unknown word extraction can also be found in the experiment of Sproat & Emerson (2003). They used the dictionary, which is composed of all words in the testing corpus, with a simple maximum matching algorithm to segment the testing corpus. They found the segmentation performance was close to perfect. Therefore, we could say that the unknown word extraction is the key technology for Chinese segmentation. In our system, the performance of unknown word detection would affect the extraction performance significantly. Although the performance of unknown word detection is acceptable, we think there is still room for improvement. Possible strategies for improving our future system include using contextual semantic relations in detection and some updated statistical methods, such as support vector machine and maximal entropy, to achieve better performance of unknown word detection.

Regarding segmentation ambiguity resolution, most of the errors are caused by covering ambiguities. The errors are caused by the heuristic Rule 1 - Longest Matching Rule - because of the occurrence of compound words, which are the composition of two words. Longest matching, or simple probabilistic, models do not solve the problem of covering ambiguities. This requires deeper context analysis.

The performance on the PK tracks was not as good as on the HK tracks. An important reason was that the coding conversion may have caused errors. For instance, in the conversion of the GB code of “里約” (the capital of Brazil) to its BIG5 codes, since GB code to BIG5 conversion is a one-to-many mapping, the above example is wrongly converted to “裡約”. This kind of error affects the accuracy of the segmentation significantly, especially for the unkown word processes. To solve this problem, we think the best and most direct solution is to re-implement the GB segmentation version according to the PK segmentation standard, without any code conversion.

References

[1] Chomsky, N. 1956, “Three models for the description of language,” *IRE Transactions on Information Theory, 2*, pages 113-124

[2] Allen James 1995 *Natural Language understanding. Second Edition*, page 44

[3] Chen, K.J. & S.H. Liu, 1992, “Word Identification for Mandarin Chinese Sentences,” *Proceedings of 14th Coling*, pages 101-107

[4] CKIP, 1993, “The Analysis of Chinese Category,” *Technical Report no. 93-05*

[5] Chen, C. J., M.H. Bai, & K.J. Chen, 1997, “Category Guessing for Chinese Unknown Words,” *Proceedings of the Natural Language Processing Pacific Rim Symposium*, pages 35-40

[6] Chen, K.J. & M.H. Bai, 1998, “Unknown Word Detection for Chinese by a Corpus-based Learning Method,” *International Journal of Computational linguistics and Chinese Language Processing*, Vol.3, #1, pages 27-44

[7] Chen, K.J.,1999, “Lexical Analysis for Chinese- Difficulties and Possible Solutions,” *Journal of Chinese Institute of Engineers*, Vol. 22. #5, pages 561-571.

[8] Church, Kenneth W., 2000, “Empirical Estimates of Adaptation: The Chance of Two Noriegas is Closer to p/2 than p*p”, *Proceedings of Coling 2000*, pages 180-186.

[9] Chen, K.J. & W.Y. Ma, 2002, “Unknown Word Extraction for Chinese Documents,” *Proceedings of COLING 2002*, pages 169-175

[10] Tseng, H.H. & K.J. Chen, 2002. “Design of Chinese Morphological Analyzer,” *Proceedings of SIGHAN*, pages 49-55

[11] Sproat, R., & Thomas E., 2003, “The first International Chinese Word Segmentation Bakeoff,” *Proceedings of SIGHAN*, pages 133-143

[12] Ma W.Y. & K.J. Chen, 2003, “A bottom-up Merging Algorithm for Chinese Unknown Word Extraction,” *Proceedings of SIGHAN*, pages 31-38

[13] Ma W.Y. & K.J. Chen, 2003, “Introduction to CKIP Chinese Word Segmentation System for the First International Chinese Word Segmentation Bakeoff,” *Proceedings of SIGHAN*, pages 168-171