Chinese Parsing in a Phoneme-to-Character Conversion System Based on Semantic Pattern Matching

Wen-Lian Hsu

Abstract

We have recently developed a Chinese phoneme-to-character conversion system with a conversion rate close to 96%. The underlying algorithm, called the context sensitive method, is based on "semantic pattern matching". The construction of these semantic patterns is largely based on linguistic common sense and corpus statistics. An interesting finding is that this method is well suited for many other types of Chinese NLP. In this paper we apply this method to the construction of a Chinese parser in the phoneme-to-character conversion system.

Keywords: semantic pattern matching, phoneme-to-character conversion, context sensitive method.

1. Introduction

A new context sensitive method (CSM) was developed for constructing a Chinese phoneme-to-character (PTC) automatic conversion system called "GOING". This method is unconventional in that it relies heavily on semantic pattern matching. Semantic patterns provide an efficient way to reduce the huge amount of data processing, which is required for homophonic character selection in Chinese phonetic input. The current conversion rate is close to 96% based on a random sampling from a corpus with seven million Chinese characters collected from Freedom Times. The major advantage of the CSM is that the conversion rate could be continuously improved without any conceivable limit.

There have been a number of approaches for the PTC system [1~12]. Besides a few attempts on using grammar or semantic analysis [4,8,9,10,11], most of these methods are based on the manipulation (for example, using dynamic programming, hidden Markov model) of word frequency and bigrams. Our approach is not just based on linguistic knowledge, but more on computational and psychological consideration. There are three dominant factors in shaping our CSM method:

1. Our goal is to develop a robust "engine" for natural language processing which is not restricted only to building a PTC system or to processing Chinese NLP. There are many approaches in NLP which are suitable only for very restricted cases. For example, an approach designed for the Chinese PTC system could very well be useless for building a text-to-speech system; a bigram table constructed for Chinese word segmentation probably could not suit the purpose of locating unknown names. In Section 5 we discuss the various systems that the CSM can be applied to). Thus, the design approach should be general enough to accommodate the differences and exceptions among various languages and systems.
2. Although a system that truly "understands" natural language is not likely to be crystallized in the near future, we only employ techniques that could shed some light towards simulating human "understanding". Thus, commonly used lower order Markov model is avoided (though the bigram and trigram statistics are collected as much as needed in perfecting our knowledge base); the usual production rules in context free languages are replaced with a large number of specific linguistic templates that incorporate both syntactic and semantic information (see [10]). As

Manuscript received on August 20, 1994; revised and accepted on November 30, 1994. The author is with the Institute of Information Science, Academia Sinica, Taipei, Taiwan, ROC.

we all know, human beings are very capable of "pattern recognition" on processing images. We believe that similar ability exists in processing natural languages by way of linguistic template matching.

3. The template construction is based largely on linguistic common sense rather than on individualized expert opinions. So that the knowledge base construction is not influenced too much by the frequent change of personnel. The idea is that we do not intend to build a system that is omnipotent in all aspects of natural language, but one that behaves more like an ordinary people. In addition, if the application of templates are scheduled systematically, then it is easier to avoid possible contradictory rules.

2. Main Idea of the CSM

Chinese words do not have delimiter markers (there are no blank space between words). Hence, traditional Chinese parsing algorithms usually start with word segmentation, followed by syntactic parsing. The semantic information is then used to weed out those parse trees that are not meaningful. In fact, much research has been concentrated on word segmentation, syntactic parsing; and relatively little has been done on semantic analysis. The problem with such an approach is that,

(1) because of the lack of syntactic and semantic information, mistakes could occur in the word segmentation stage. These errors might propagate into later analysis and produce an incorrect parse tree. This problem becomes more prominant when dealing with phoneme sequences.

(2) traditional syntactic parsing algorithm based on the context free model is very inefficient (a cubic algorithm) and could result in quite a number of possible parse trees, which creates a big computational burden as well as nondeterminism. In the CSM, many semantic templates are set up initially so that any sentence must match a large number of these templates in order to be legal, which often results in a unique parse tree.

The philosophy of the CSM is to simulate human thinking based on "template matching". We use templates to model contextual information. The construction of these templates is largely based on common sense and statistics. Furthermore, probabilities (collected based on a large corpus) are assigned to templates so that the final outcome is not decided by a fixed set of rules (as in many traditional rule-based system); rather, the accumulated effect of different matched templates makes the desired outcome more likely to be obtained. The design of the CSM is intended to resolve local ambiguities using as much contextual information as possible.

Computational linguistic models often suffers from the fact that there are many different approaches for treating different aspects of language processing and therefore, it is difficult to combine them together. The underlying principle of the CSM is purely computational and bears no relation on the linguistic knowledge per se. Thus, the main feature of the CSM is its ability to accommodate all kinds of linguistic knowledge as well as computational efficiency consideration in a coherent fashion. The robustness of the CSM has already shown its effectiveness in the PTC system "GOING". We shall discuss the application of the CSM in constructing a parser for the PTC system.

It is difficult enough to "correctly" parse a Chinese sentence given the characters and much harder to obtain a correct parse tree given only the phonemes in an automatic PTC system. The existence of a large number of homophonic characters (normally five to fifteen characters corresponding to one phoneme) could create an exponential number of possible sentences. Even if we eliminate some of these sentences based on multicharacter words, the possible combinations is still enormous. A question naturally arises is "Why are we interested in Chinese parsing given only the phonemes knowing that parsing given the characters is difficult enough?" The reason is that we are interested in a more general objective -- natural language understanding. Although the existence of a large number of homophonic characters makes parsing difficult, it does not create much verbal communication problem among Chinese people, even for those with less education. Thus, rigorous grammatical training does not seem to be a prerequisite for understanding Chinese. Based on these observations, we make the following assumption about "human understanding".

The main tool for human understanding and reasoning is through "**semantic pattern matching**"

Many of these patterns are constructed out of common sense. Pattern matching is not only our way of interpreting human "instinct", it is also pertaining to the way we describe logical inferences. We believe that human thinking process can be modeled as "abstract" templates, though the exact form is not yet explicitly known. This fact is clearly illustrated in the following phenomena: (1) when taking exams, most students are simply doing lower order pattern matching from problems to solution techniques with very little time to think from scratch; (2) brilliant mathematicians usually can react very fast in theoretical discussion (perhaps because they have more higher order templates to match the subject swiftly).

3. Semantic Pattern Matching of the CSM

The semantic pattern matching of the CSM is based on a large collection of "templates". Each template is a description of the occurrence of a word or a collection of words. The structure of a template is a hybrid combination of syntactic and semantic features. Such a description could be either local or global. The matching of the templates are divided into stages so that the more local ones will be applied before the more general ones. The CSM parser does not employ a simple production system. Rather, the establishment of legal sentences is based on the successful matching of a series of word, phrase and sentence templates. Word segmentation, syntactic analysis and semantic analysis which are traditionally divided into three stages are now carried out simultaneously in the CSM.

Some examples of templates are given below for the PTC system. In Chinese, different measures are used in accordance with the nouns they specified. Consider the following sentences, "A very lovely cat", "A very lovely flower" and "A very lovely candle". The same quantifier "a" is used with all three nouns in three English sentences, but their counterparts in Chinese are all different.

a very lovely ( )
一 隻 很 可 愛 的 貓
yi4 zhi1 heng3 ke3 ai4 de5 mou1 (cat)
{ determinative | 隻 (zhi1) | adv. | adj. | 的 | noun[animal] }

一 枝 很 可 愛 的 花
yi4 zhi1 heng3 ke3 ai4 de5 hua1 (flower)
{ determinative | 枝 (zhi1) | adv. | adj. | 的 | noun[plant] }

一 支 很 可 愛 的 蠟 燭
yi4 zhi1 heng3 ke3 ai4 de5 la4 chu2 (candle)
{ determinative |支 (zhi1) | adv. | adj. | 的 | noun[artifact] }

Since word frequency statistics favors only one of these measures (which will be selected regardless of the semantic category of the nouns), it will produce absurd results in the other cases. In contrast, the use of appropriate templates (illustrated below each of the above sentences) can effectively identify the correct measure provided that the category of nouns are specified in detail. By the way, the above example is even more complex than it appears to be: there are, in addition, 70 homophonic characters corresponding to the phoneme "yi4" and 10 homophonic characters for the phoneme "zhi1". Furthermore, "yi4 zhi1" also corresponds to the word "artificial limb".

The above templates not only are useful for identifying the correct measure in the PTC system, they also serve to identify the NPs. This is quite important in CSM parsing. The parser for the CSM works in stages: it uses local word or phrase templates to locate possible words, NPs, VPs and APs, and then use the sentence templates to identify the final sentence structure. Although there could be ambiguities in identifying the correct words, NPs or VPs, the semantics of matched templates restricts the number of possible candidates to a manageable size. In fact, specific templates can be constructed to make the final parse tree unique in most cases. We illustrate the four stages of template matching in Figure 1.

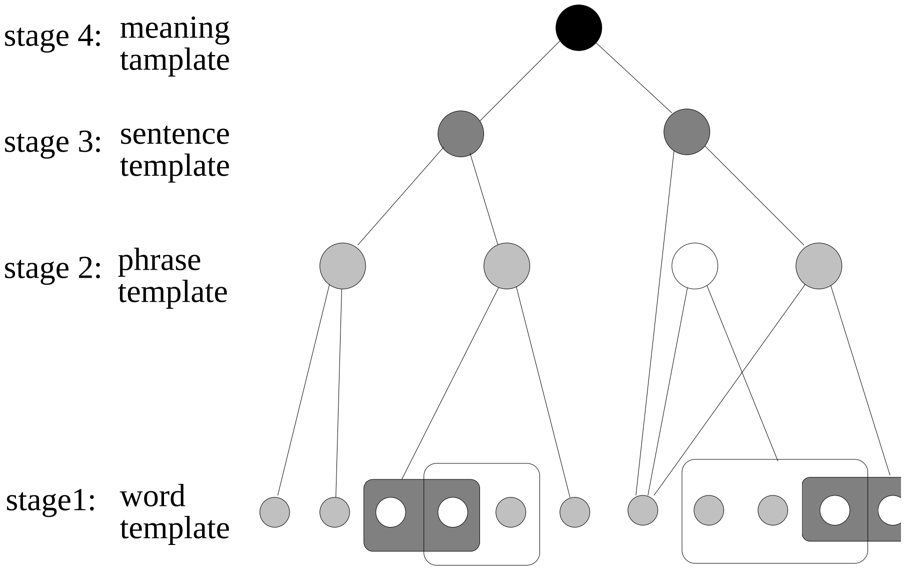

Figure 1. The hierarchy of template matching

The bottom nodes of Figure 1 gives the sequence of input phonemes. The rectangles enclosing some of the phonemes give candidate words (for clarity, we omit the character candidates). Hence, two overlapping rectangles indicate the overlapping of their corresponding words. Instead of breaking the ties using frequencies, the CSM resolves the word boundary ambiguity by considering higher level structural templates, such as phrase templates or sentence templates. The four nodes in the phrase template level indicate that it is possible to form four phrases from the bottom phonemes. However, as indicated, only three of these four phrase templates will give rise to meaningful sentence templates (the one phrase template that does not form part of legal sentence templates is denoted by the white node). Finally, assuming there are templates that gives rise to the representation of meaning (or understanding). Then the ultimate goal of the CSM is to obtain a meaning template.

The CSM can be considered as some kind of relaxation; namely, allow all possible candidates to be present initially, and then gradually discover the appropriate ones through higher order template matching. The key issue resides in the construction of a good collection of templates.

Now, let us consider these stages of template matching operations separately since full parsing is not required in some NLP applications. If only the first stage is applied, those component words with the largest accumulated weights will remain in the final result. In this case, the order of application of the templates becomes crucial, since we want those templates with larger weights to be applied before those with lower weights; also, care must be taken to deal with the problem of conflicting templates by assigning them with appropriate weights. The danger of not applying the higher order templates is that the decision made based on more local structure could very well be flawed. However, for many applications, sentence templates are used only sparingly since satisfactory results are already within reach based on lower level phrase and word templates.

3. Template construction and searching

In this section we discuss what kind of templates need to be collected in the CSM parser and how to construct them. In principle, the occurrences of every word in Chinese NL can be described by a collection of templates; namely, one can exhaustively list all possible ways that such a word can be used legally and represent them abstractly by semantic templates. This may appear to be a formidable task. However, it is not much different from the way one learns how to use this word correctly. Statistically speaking, one needs to record only those few templates that occur most frequently and illuminate the usage of this word. Also, similar templates can be grouped together and share the same form so that a more general template can be constructed to replace this group. All of these make the template collecting task much more tractable and the database size more manageable.

We claim that, the number of templates required in the CSM is not much more than what ordinary people possess. Considering the amount of knowledge required for a person to process Chinese NL, the template collecting task seems quite natural. In fact, template matching is prevalent in all languages. For example, consider the following collection of synonyms in English,

beautiful
good looking
gorgeous
splendid
magnificent
great

Although these words sometimes can be used interchangeably, each of them has its own special identity. Most high school students would be able to correctly use these words in composition. Our explanation is that the students possess something equivalent to the collection of templates needed to disambiguate the meaning of these synonyms. In the CSM, the use of these synonyms in different "contextual" situations are recorded in templates. Below we shall discuss several types of templates each of which serves to disambiguate certain sentence patterns.

In the following sentences, the phoneme sequences for "I hire him as" and "My tenure is" are exactly the same in Chinese. We need to use two templates associated with "hire" and "tenure" to disambiguate them.

我　聘　其　為　教授
I　hire　he　as　professor
(I hire him as a professor)
{ N[person] | hire | N[person] | as | N[occupation] }

我　聘期　為　一　年
I　tenure　is　one　year
(My tenure is for one year)
{ N[duration] | VP | N[time] }

With these two templates we can easily distinguish the two homophonic sequences, and the word "tenure", which is longer than "hire" in Chinese, will not overtake the shorter word "hire" all the time. In the next example, one needs to identify the object correctly.

他　趕走了　那隻　村裡 的　人　都　討厭 的　狗。
he　drive away　that　village　people　all　hate　dog
(He drove away the dog that all village people hated)
The template to be used for NP[dog] is:
{ determinater | 隻 ∥ 的 | N[animal] }
(where "∥" stands for "not necessarily neighboring")

This template will identify the noun phrase

那隻村裡的人都討厭的狗.

(the dog that all village people hated)

So we are left with

他 趕走了 | NP[dog]

he drive away | NP[dog]

Then the whole sentence can be easily parsed by the sentence template with the key on the main verb "drive away" (趕走了). In case there is an ambiguity about the NP, other templates can be used to disambiguate it. The following example illustrates this:

一隻 附近的 狗 都 討厭的 野狗 在偷吃 我家的 魚

a neighboring dog all disliked wild dog is eating my family fish

(A wild dog that is disliked by all neighboring dogs is eating my fish)

There is a possible ambiguity about the main verb and the two NPs: "a wild dog" or "a neighboring dog". Such an ambiguity can be resolved by constructing the following template on the word "all", which is a template for a modifier:

{ 名詞 | 都 | 動詞 | 的 | 名詞 }

{ S[NP | all | VP] | de5 | NP }

This template indicates that it is very likely that "neighboring dogs" is the subject of the clause modifying the "wild dog". Since there is no template supporting the alternative "A neighboring dog" to be the main subject, the correct parsing can be obtained.

In the following we further consider some examples in which the change of a single word could alter the sentence structure completely.

(1) 他 說 你 很 有趣。

he say you very funny

(He said that you are very funny)

(2) 他 說話 很 有趣。

he talk very funny

(He talks very funny.)

(3) 小丑 逗 猴子 很 有趣。

clown play monkey very funny

(That the clown plays with the monkey is very funny.)

(4) 小丑 逗 猴子 很 開心。

clown play monkey very happy

(The clown plays with the monkey very happily.)

In (1), (3) and (4) the sentence structures in Chinese are the same

NP - VP - NP - VP

Although they all end up with the same VP "very funny", the parse trees are quite different. The only difference between (3) and (4) is the last word, where "funny" is replaced by "happy". But that changes the meaning of the two sentences completely. Next, we consider changing (2) a little to

(2') 他 說 這 話 很 有趣

he say this word very funny

then a possible meaning would be

(The fact that he said this is very funny)

which is entirely different from (2). The only way to avoid the above ambiguity is to specify the different contextual situations in which the words "funny" and "happy" (or two larger semantic categories separating these two words) appear. All of these indicate that the semantics of words are absolutely essential in Chinese parsing. Below, we illustrate a few sentence templates that could disambiguate sentences (1), (3) and (4):

(i) { 人 | 說 | S[NP[人] | VP] }

{person say S[NP[person] | VP] }

(ii) { S | 很 | 有趣 }

{ S | adverb | funny}

(iii) { 人 | VP[active] | NP | 很 | 開心 }

{ person VP[active] NP adverb happy}

Philosophically speaking, we can construct as many templates as needed to resolve sentence ambiguity as long as these templates are not contradictory to each other. In practice, it pays to select those few templates that are more effective. Statistically, we can count which template is used more often and increase its strength. For example, when both (i) and (ii) are matched with the target sentence, we found (i) is more often the correct one; hence, (i) should receive a larger weight (or probability). A more general template should usually receive less weight than the more specific ones. Of course, there are other ways to help decide which template is more likely in a particular context using higher order templates, for example, discourse analysis. Also, if some template is creating a lot of conflicts with others, we can always replace it with more restricted ones. Thus, the following would not be suitable as a general template since it is highly ambiguous:

NP - VP - NP - VP

5. Implementation

As we can see from the previous sections, templates are constructed basically to disambiguate among possible candidate characters or words. So for a parser to be really comprehensive a lot of templates need to be constructed. In the construction of "GOING" for the PTC system we have approximately 50,000 word templates and 30,000 semantic phrase templates. These templates are certainly not sufficient for a Chinese parser. However, in between the status quo and the final ideal template collection, we shall gradually add to the existing class certain general sentence templates, which are easily available in many linguistic books. These templates will ensure that most of the legal sentences will also be matched under the current template system. Additional templates can be constructed as situation warrants. For the time being, we shall only consider the application of semantic templates to the PTC system.

In the actual implementation of the PTC system, parsing is used only as a supporting mechanism. When parsing is unsuccessful, lower order templates will start voting (namely, adding weights) for its components and such a voting strategy often provides excellent result for the PTC system. That is because there are a lot of semantic templates obtained at the phrase level or even the word level, which are good enough to disambiguate the homophonic characters.

Computational linguists who disbelieve the effectiveness of any rule based system would probably wonder: "How can one build a large template database without confusing oneself?" We shall try to answer this using the following guidelines for practical template construction:

(1) These templates should be as specific as possible so that the probability of having two conflicting templates is very small (even when that happens conflict resolution is much easier). We found that many regular grammar rules

are often not useful in template construction simply because there are too many exceptions. However, in most cases, these grammar rules can be further refined to generate useful semantic templates.

(2) The types of different templates should be as limited as possible so that those who are constructing new templates can easily understand exactly how they fit within the original collection of templates.

(3) The effect of a template should be calculated in a probabilistic sense. Statistical tools which evaluate the effectiveness of each template based on a large corpus should be constructed so that the marginal value of the incoming one will be obvious.

(4) After a period of template construction, there should be a reorganizing tool to detect the possible conflicts, redundancy, and priority change among the existing templates.

(5) The schedule of applying different types of templates should be carefully arranged and conflicting templates need to be resolved by changing their associated weights based on a more global guideline.

Many of the guidelines are based on the consideration of computational issues rather than on the linguistic ones. They have been applied successfully in our GOING system. Templates are normally indexed on one (or more) of its "key" components. Our current PTC system consists of 50,000 words and more than 30,000 templates, but only a small portion is actually used in matching for each input sentence because of the careful selection of the "keys". The total space occupied by the knowledge database is around 500K. On an IBM 386 AT compatible machines, it takes about 0.1 second to convert one sentence. To construct a parser for the PTC system many more templates are needed (roughly four or five more times our current size). However, because the complexity of the CSM is almost linearly dependent on the number of templates, a constant increase on the template number does not create any computational problem. We believe that templates are the most efficient ways to extract the contextual information.

Given a large corpus, one can easily collect various word frequencies or collocational probabilities. However, it is much harder to collect the statistics for semantic related structures. Semantic linking normally associates word-senses, not the actual words or characters. Also, semantic information can often relate words that are farther apart from each other, which requires more intelligent statistical methods to detect.

6. Other applications of the CSM

As mentioned before, the idea of building a template matching system can be easily applied to many other NLP systems. We shall discuss a few below.

(1) Chinese spelling checker

Unlike an English spelling checker, which can be easily built on a dictionary and a simple approximate matching algorithm, Chinese spelling checker requires much deeper analysis (since words do not have delimiter markers). Of course, basic parsing is required, but the difficult issue is to deal with possible anticipation from the context. We believe that templates can be used to model "expectation"--- if the majority of components of a template have appeared, then one could "expect" the remaining ones should also occur.

(2) Machine translation

The CSM template construction can be applied to any language to obtain a parser. Thus, in English to Chinese translation, one can first collect templates in English and then relate them to the corresponding templates in Chinese. Translation can be carried out through higher order template (phrase and sentence templates) transformation.

(3) Text-to-speech conversion

In a text-to-speech system we need to transfer Chinese characters to appropriate phonemes as a first step. To produce more natural intonation several parameters (e.g.. pitch, length and volume) must be set appropriately. All of these are heavily context dependent. Thus, the template model representing the contextual information can be used.

Acknowledgment

We would like to thank Dr. K. J. Chen for his participation in the earlier part of the PTC system project. We are also grateful for all those people who have participated in this project during the past five years.

References

1. Kuo, J.J., et al., "The development of New Chinese Input Method - Chinese Word String Input Method," Proceedings of International Computer Symposium, Taipei, (1986).
2. Chen, S.I. et al., "The Continuous Conversion Algorithm of Chinese Character's Phonetic Symbols to Chinese Characters, "Proceedings of National Computer Symposium, Taipei, (1986).
3. Fan, C.K. and W.H. Tsai, "Disambiguation of Phonetic Chinese Input by Relaxation-based Word Identification," Proceedings of ROCLING I, (1988), 145-160.
4. Hsieh, M.L., T.T. Lo and C.H. Lin, "Grammatical Approach to Converting Phonetic Symbols into Characters," Proceedings of National Computer Symposium, Taipei, (1989), 453-461.
5. Sproat, R., "An Application of Statistical Optimization with Dynamic Programming to Phonemic-Input-to-Character Conversion for Chinese, " Proceedings of ROCLING III, (1990), 379-390.
6. Gu, H.Y., C.Y. Tseng and L.S. Lee, "Markov Modeling of Chinese Language for Linguistically Decoding the Mandarin Phonetic Input, Proceedings of National Computer Symposium, Taipei, (1989), 759-767.
7. Chang, J.S., S.D. Chern and C.D. Chen, "Conversion of Phonemic-Input to Chinese Text Through Constraint Satisfaction," Proceedings of ROCKLING IV, (1991), 30-36.
8. Lua, K.T. and K.W. Gan, "A Touch-Typing Pinyin Input System," Computer Processing of Chinese and Oriental Languages 6, (1992), 85-94.
9. Gie, T.H., "A Phonetic Input System for Chinese Characters Using a Word Dictionary and Statistics," Master Thesis, National Taiwan University, 1991.
10. 許聞廉、陳克健，"「國音」智慧型輸入系統的語意分析——「脈絡會意法」，" 第二屆計算語言學聯合學術會議論文集，廈門，(1993)，338-343.
11. 龐民治、張永光，"獨立於輸入法的漢字輸入模型及利用中文自然語言知識後處理的漢字輸入，" 第二屆計算語言學聯合學術會議論文集，廈門，(1993)，344-349.
12. Chen, K.J., "A Mathematical Model for Chinese Input," Computer Processing of Chinese and Oriental Languages 7, 75-84.

The fact that it is almost effortless for ordinary Chinese to select the appropriate homophones in conversation (certainly required for understanding) forces us to believe there is a more robust model for understanding.　Of course, we cannot assume ordinary Chinese possess sophisticated linguistic knowledge nor can we assume they are capable of swift logical derivation.

The only straightforward explanation we could imagine is that people understand conversational Chinese simply through semantic pattern matching.　Furthermore, most of these patterns are constructed out of common sense.