Parsing the Internal Structure of Words: A New Paradigm for Chinese Word Segmentation

Zhongguo Li
State Key Laboratory on Intelligent Technology and Systems
Tsinghua National Laboratory for Information Science and Technology
Department of Computer Science and Technology
Tsinghua University, Beijing 100084, China
eemath@gmail.com

Abstract

Lots of Chinese characters are very productive in that they can form many structured words either as prefixes or as suffixes. Previous research in Chinese word segmentation mainly focused on identifying only the word boundaries without considering the rich internal structures of many words. In this paper we argue that this is unsatisfying in many ways, both practically and theoretically. Instead, we propose that word structures should be recovered in morphological analysis. An elegant approach for doing this is given and the result is shown to be promising enough for encouraging further effort in this direction. Our probability model is trained with the Penn Chinese Treebank and actually is able to parse both word and phrase structures in a unified way.

1 Why Parse Word Structures?

Research in Chinese word segmentation has progressed tremendously in recent years, with state of the art performing at around 97% in precision and recall (Xue, 2003; Gao et al., 2005; Zhang and Clark, 2007; Li and Sun, 2009). However, virtually all these systems focus exclusively on recognizing the word boundaries, giving no consideration to the internal structures of many words. Though it has been the standard practice for many years, we argue that this paradigm is inadequate both in theory and in practice, for at least the following four reasons.

The first reason is that if we confine our definition of word segmentation to the identification of word boundaries, then people tend to have divergent opinions as to whether a linguistic unit is a word or not (Sproat et al., 1996). This has led to many different annotation standards for Chinese word segmentation. Even worse, this could cause inconsistency in the same corpus. For instance, 副主席 'vice president' is considered to be one word in the Penn Chinese Treebank (Xue et al., 2005), but is split into two words by the Peking University corpus in the SIGHAN Bakeoffs (Sproat and Emerson, 2003). Meanwhile, 副导演 'vice director' and 副经理 'deputy manager' are both segmented into two words in the same Penn Chinese Treebank. In fact, all these words are composed of the prefix 副 'vice' and a root word. Thus the structure of 副主席 'vice president' can be represented with the tree in Figure 1. Without a doubt, there is complete agreement on the correctness of this structure among native Chinese speakers. So if instead of annotating only word boundaries, we annotate the structures of every word, [1] then the annotation tends to be more

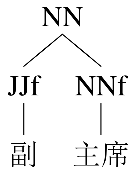

Figure 1: Example of a word with internal structure.

[1] Here it is necessary to add a note on terminology used in this paper. Since there is no universally accepted definition of the "word" concept in linguistics and especially in Chinese, whenever we use the term "word" we might mean a linguistic unit such as 副主席 'vice president' whose structure is shown as the tree in Figure 1, or we might mean a smaller unit such as 主席 'president' which is a substructure of that tree. Hopefully,

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 1405–1414,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

consistent and there could be less duplication of efforts in developing the expensive annotated corpus.

The second reason is applications have different requirements for granularity of words. Take the personal name 周树人 ‘Zhou Shuren’ as an example. It’s considered to be one word in the Penn Chinese Treebank, but is segmented into a surname and a given name in the Peking University corpus. For some applications such as information extraction, the former segmentation is adequate, while for others like machine translation, the later finer-grained output is more preferable. If the analyzer can produce a structure as shown in Figure 4(a), then every application can extract what it needs from this tree. A solution with tree output like this is more elegant than approaches which try to meet the needs of different applications in post-processing (Gao et al., 2004).

The third reason is that traditional word segmentation has problems in handling many phenomena in Chinese. For example, the telescopic compound 大中小学 ‘universities, middle schools and primary schools’ is in fact composed of three coordinating elements 大学 ‘university’, 中学 ‘middle school’ and 小学 ‘primary school’. Regarding it as one flat word loses this important information. Another example is separable words like 游泳 ‘swim’. With a linear segmentation, the meaning of ‘swimming’ as in 游完泳 ‘after swimming’ cannot be properly represented, since 游泳 ‘swim’ will be segmented into discontinuous units. These language usages lie at the boundary between syntax and morphology, and are not uncommon in Chinese. They can be adequately represented with trees (Figure 2).

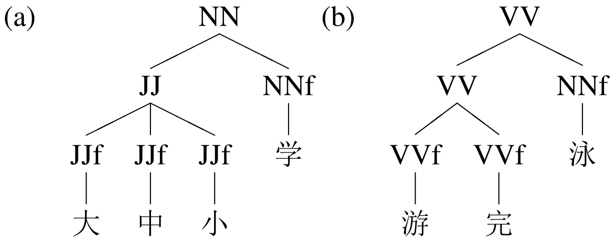

Figure 2: Example of telescopic compound (a) and separable word (b).

The last reason why we should care about word structures is related to head driven statistical parsers (Collins, 2003). To illustrate this, note that in the Penn Chinese Treebank, the word 英格兰人 ‘English People’ does not occur at all. Hence constituents headed by such words could cause some difficulty for head driven models in which out-of-vocabulary words need to be treated specially both when they are generated and when they are conditioned upon. But this word is in turn headed by its suffix 人 ‘people’, and there are 2,233 such words in Penn Chinese Treebank. If we annotate the structure of every compound containing this suffix (e.g. Figure 3), such data sparsity simply goes away.

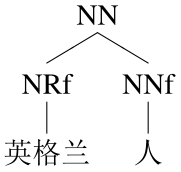

Figure 3: Structure of the out-of-vocabulary word 英格兰人 ‘English People’.

Had there been only a few words with internal structures, current Chinese word segmentation paradigm would be sufficient. We could simply recover word structures in post-processing. But this is far from the truth. In Chinese there is a large number of such words. We just name a few classes of these words and give one example for each class (a dot is used to separate roots from affixes):

personal name: 长尾·真 ‘Nagao Makoto’
location name: 纽约·州 ‘New York State’
noun with a suffix: 分类·器 ‘classifier’
noun with a prefix: 准·妈妈 ‘mother-to-be’
verb with a suffix: 自动·化 ‘automatize’
verb with a prefix: 防·水 ‘waterproof’
adjective with a suffix: 复合·型 ‘composite’
adjective with a prefix: 非·正式 ‘informal’
pronoun with a prefix: 各·位 ‘everybody’
time expression: 一九九五·年 ‘the year 1995’
ordinal number: 第·十一 ‘eleventh’
retroflex suffixation: 花朵·儿 ‘flower’

This list is not meant to be complete, but we can get a feel of how extensive the words with non-trivial structures can be. With so many productive suffixes and prefixes, analyzing word structures in post-processing is difficult, because a character may or may not act as an affix depending on the context.

the context will always make it clear what is being referred to with the term “word”.

For example, the character 人 'people' in 植树人 'the one who plants' is a suffix, but in the personal name 周树人 'Zhou Shuren' it isn't. The structures of these two words are shown in Figure 4.

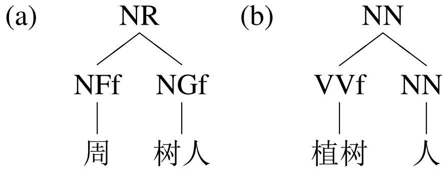

Figure 4: Two words that differ only in one character, but have different internal structures. The character 人 'people' is part of a personal name in tree (a), but is a suffix in (b).

A second reason why generally we cannot recover word structures in post-processing is that some words have very complex structures. For example, the tree of 无政府主义者 'anarchist' is shown in Figure 5. Parsing this structure correctly without a principled method is difficult and messy, if not impossible.

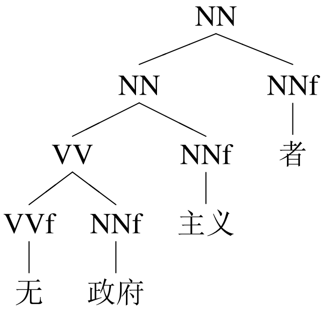

Figure 5: An example word which has very complex structures.

Finally, it must be mentioned that we cannot store all word structures in a dictionary, as the word formation process is very dynamic and productive in nature. Take 馆 'hall' as an example. Standard Chinese dictionaries usually contain 图书馆 'library', but not many other words such as 海洋馆 'aquarium' generated by this same character. This is understandable since the character 馆 'hall' is so productive that it is impossible for a dictionary to list every word with this character as a suffix. The same thing happens for natural language processing systems. Thus it is necessary to have a dynamic mechanism for parsing word structures.

In this paper, we propose a new paradigm for Chinese word segmentation in which not only word boundaries are identified but the internal structures of words are recovered (Section 3). To achieve this, we design a joint morphological and syntactic parsing model of Chinese (Section 4). Our generative story describes the complete process from sentence and word structures to the surface string of characters in a top-down fashion. With this probability model, we give an algorithm to find the parse tree of a raw sentence with the highest probability (Section 5). The output of our parser incorporates word structures naturally. Evaluation shows that the model can learn much of the regularity of word structures, and also achieves reasonable accuracy in parsing higher level constituent structures (Section 6).

2 Related Work

The necessity of parsing word structures has been noticed by Zhao (2009), who presented a character-level dependency scheme as an alternative to the linear representation of words. Although our work is based on the same notion, there are two key differences. The first one is that part-of-speech tags and constituent labels are fundamental for our parsing model, while Zhao focused on unlabeled dependencies between characters in a word, and part-of-speech information was not utilized. Secondly, we distinguish explicitly the generation of flat words such as 华盛顿 'Washington' and words with internal structures. Our parsing algorithm also has to be adapted accordingly. Such distinction was not made in Zhao's parsing model and algorithm.

Many researchers have also noticed the awkwardness and insufficiency of current boundary-only Chinese word segmentation paradigm, so they tried to customize the output to meet the requirements of various applications (Wu, 2003; Gao et al., 2004). In a related research, Jiang et al. (2009) presented a strategy to transfer annotated corpora between different segmentation standards in the hope of saving some expensive human labor. We believe the best solution to the problem of divergent standards and requirements is to annotate and analyze word structures. Then applications can make use of these structures according to their own convenience.

Since the distinction between morphology and syntax in Chinese is somewhat blurred, our model for word structure parsing is integrated with constituent parsing. There has been many efforts to integrate Chinese word segmentation, part-of-speech tagging and parsing (Wu and Zixin, 1998; Zhou and Su, 2003; Luo, 2003; Fung et al., 2004). However, in these research all words were considered to be flat, and thus word structures were not parsed. This is a crucial difference with our work. Specifically, consider the word 橄榄油 'olive oil'. Our parser output tree Figure 6(a), while Luo (2003) output tree (b), giving no hint to the structure of this word since the result is the same with a real flat word 洛杉矶 'Los Angeles'(c).

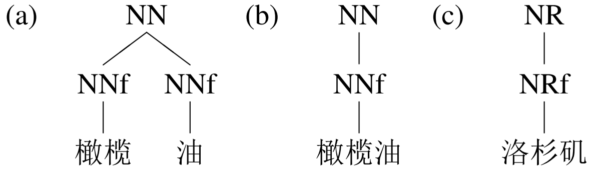

Figure 6: Difference between our output (a) of parsing the word 橄榄油 'olive oil' and the output (b) of Luo (2003). In (c) we have a true flat word, namely the location name 洛杉矶 'Los Angeles'.

The benefits of joint modeling has been noticed by many. For example, Li et al. (2010) reported that a joint syntactic and semantic model improved the accuracy of both tasks, while Ng and Low (2004) showed it's beneficial to integrate word segmentation and part-of-speech tagging into one model. The later result is confirmed by many others (Zhang and Clark, 2008; Jiang et al., 2008; Kruengkrai et al., 2009). Goldberg and Tsarfaty (2008) showed that a single model for morphological segmentation and syntactic parsing of Hebrew yielded an error reduction of 12% over the best pipelined models. This is because an integrated approach can effectively take into account more information from different levels of analysis.

Parsing of Chinese word structures can be reduced to the usual constituent parsing, for which there has been great progress in the past several years. Our generative model for unified word and phrase structure parsing is a direct adaptation of the model presented by Collins (2003). Many other approaches of constituent parsing also use this kind of head-driven generative models (Charniak, 1997; Bikel and Chiang, 2000) .

3 The New Paradigm

Given a raw Chinese sentence like 林志浩是总工程师, a traditional word segmentation system would output some result like 林志浩 是 总工程师('Lin Zhihao', 'is', 'chief engineer'). In our new paradigm, the output should at least be a linear sequence of trees representing the structures of each word as in Figure 7.

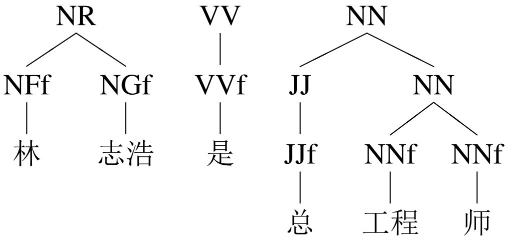

Figure 7: Proposed output for the new Chinese word segmentation paradigm.

Note that in the proposed output, all words are annotated with their part-of-speech tags. This is necessary since part-of-speech plays an important role in the generation of compound words. For example, 者 'person' usually combines with a verb to form a compound noun such as 设计者 'designer'.

In this paper, we will actually design an integrated morphological and syntactical parser trained with a treebank. Therefore, the real output of our system looks like Figure 8. It's clear that besides all the information of the proposed output for the new

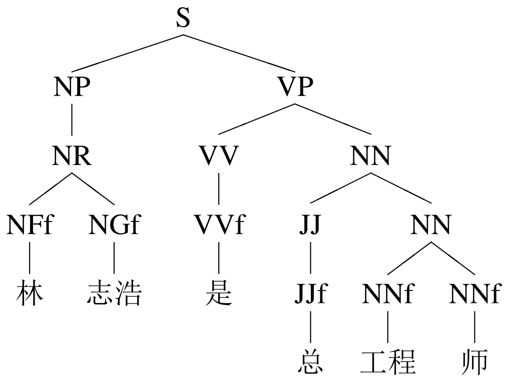

Figure 8: The actual output of our parser trained with a fully annotated treebank.

paradigm, our model's output also includes higher-level syntactic parsing results.

3.1 Training Data

We employ a statistical model to parse phrase and word structures as illustrated in Figure 8. The currently available treebank for us is the Penn Chinese Treebank (CTB) 5.0 (Xue et al., 2005). Because our model belongs to the family of head-driven statistical parsing models (Collins, 2003), we use the head-finding rules described by Sun and Jurafsky (2004).

Unfortunately, this treebank or any other treebanks for that matter, does not contain annotations of word structures. Therefore, we must annotate these structures by ourselves. The good news is that the annotation is not too complicated. First, we extract all words in the treebank and check each of them manually. Words with non-trivial structures are thus annotated. Finally, we install these small trees of words into the original treebank. Whether a word has structures or not is mostly context independent, so we only have to annotate each word once.

There are two noteworthy issues in this process. Firstly, as we'll see in Section 4, flat words and non-flat words will be modeled differently, thus it's important to adapt the part-of-speech tags to facilitate this modeling strategy. For example, the tag for nouns is NN as in 伊拉克 'Iraq' and 前总统 'former president'. After annotation, the former is flat, but the later has a structure (Figure 9). So we change the POS tag for flat nouns to NNf, then during bottom up parsing, whenever a new constituent ending with 'f' is found, we can assign it a probability in a way different from a structured word or phrase.

Secondly, we should record the head position of each word tree in accordance with the requirements of head driven parsing models. As an example, the right tree in Figure 9 has the context free rule "NN → JJf NNf", the head of which should be the rightmost NNf. Therefore, in 前总统 'former president' the head is 总统 'president'.

In passing, the readers should note the fact that in Figure 9, we have to add a parent labeled NN to the flat word 伊拉克 'Iraq' so as not to change the context-free rules contained inherently in the original treebank.

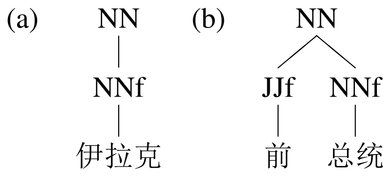

Figure 9: Example word structure annotation. We add an 'f' to the POS tags of words with no further structures.

4 The Model

Given an observed raw sentences S, our generative model tells a story about how this surface sequence of Chinese characters is generated with a linguistically plausible morphological and syntactical process, thereby defining a joint probability $\Pr(T, S)$ where T is a parse tree carrying word structures as well as phrase structures. With this model, the parsing problem is to search for the tree T^* such that

$$T^* = \arg\max_T \Pr(T, S) \tag{1}$$

The generation of S is defined in a top down fashion, which can be roughly summarized as follows. First, the lexicalized constituent structures are generated, then the lexicalized structure of each word is generated. Finally, flat words with no structures are generated. As soon as this is done, we get a tree whose leaves are Chinese characters and can be concatenated to get the surface character sequence S.

4.1 Generation of Constituent Structures

Each node in the constituent tree corresponds to a lexicalized context free rule

$$P \rightarrow L_n \, L_{n-1} \cdots L_1 H R_1 \, R_2 \cdots R_m \tag{2}$$

where P, L_i, R_i and H are lexicalized nonterminals and H is the head. To generate this constituent, first P is generated, then the head child H is generated conditioned on P, and finally each L_i and R_j are generated conditioned on P and H and a distance metric. This breakdown of lexicalized PCFG rules is essentially the Model 2 defined by Collins (1999). We refer the readers to Collins' thesis for further details.

4.2 Generation of Words with Internal Structures

Words with rich internal structures can be described using a context-free grammar formalism as

$$\text{word} \rightarrow \text{root} \tag{3}$$
$$\text{word} \rightarrow \text{word suffix} \tag{4}$$
$$\text{word} \rightarrow \text{prefix word} \tag{5}$$

Here the root is any word without interesting internal structures, and the prefixes and suffixes are not limited to single characters. For example, 主义 'ism' as in 现代主义 'modernism' is a well known and very productive suffix. Also, we can see that rules (4) and (5) are recursive and hence can handle words with very complex structures.

By (3)–(5), the generation of word structures is exactly the same as that of ordinary phrase structures. Hence the probabilities of these words can be defined in the same way as higher level constituents in (2). Note that in our case, each word with structures is naturally lexicalized, since in the annotation process we have been careful to record the head position of each complex word.

As an example, consider a word $w = R(r)\,S(s)$ where R is the root part-of-speech headed by the word r, and S is the suffix part-of-speech headed by s. If the head of this word is its suffix, then we can define the probability of w by

$$\Pr(w) = \Pr(S, s) \cdot \Pr(R, r|S, s) \tag{6}$$

This is equivalent to saying that to generate w, we first generate its head $S(s)$, then conditioned on this head, other components of this word are generated. In actual parsing, because a word always occurs in some contexts, the above probability should also be conditioned on these contexts, such as its parent and the parent's head word.

4.3 Generation of Flat Words

We say a word is flat if it contains only one morpheme such as 伊拉克 'Iraq', or if it is a compound like 开发 'develop' which does not have a productive component we are currently interested in. Depending on whether a flat word is known or not, their generative probabilities are computed also differently. Generation of flat words seen in training is trivial and deterministic since every phrase and word structure rules are lexicalized.

However, the generation of unknown flat words is a different story. During training, words that occur less than 6 times are substituted with the symbol UNKNOWN. In testing, unknown words are generated after the generation of symbol UNKNOWN, and we define their probability by a first-order Markov model. That is, given a flat word $w = c_1 c_2 \cdots c_n$ not seen in training, we define its probability conditioned with the part-of-speech p as

$$\Pr(w|p) = \prod_{i=1}^{n+1} \Pr(c_i|c_{i-1}, p) \tag{7}$$

where c_0 is taken to be a START symbol indicating the left boundary of a word and c_{n+1} is the STOP symbol to indicate the right boundary. Note that the generation of w is only conditioned on its part-of-speech p, ignoring the larger constituent or word in which w occurs.

We use a back-off strategy to smooth the probabilities in (7):

$$\begin{aligned}\tilde{\Pr}(c_i|c_{i-1}, p) &= \lambda_1 \cdot \hat{\Pr}(c_i|c_{i-1}, p) \\ &\quad + \lambda_2 \cdot \hat{\Pr}(c_i|c_{i-1}) \\ &\quad + \lambda_3 \cdot \hat{\Pr}(c_i)\end{aligned} \tag{8}$$

where $\lambda_1 + \lambda_2 + \lambda_3 = 1$ to ensure the conditional probability is well formed. These λs will be estimated with held-out data. The probabilities on the right side of (8) can be estimated with simple counts:

$$\hat{\Pr}(c_i|c_{i-1}, p) = \frac{\text{COUNT}(c_{i-1}c_i, p)}{\text{COUNT}(c_{i-1}, p)} \tag{9}$$

The other probabilities can be estimated in the same way.

4.4 Summary of the Generative Story

We make a brief summary of our generative story for the integrated morphological and syntactic parsing model. For a sentence S and its parse tree T, if we denote the set of lexicalized phrase structures in T by $\mathcal{C}$, the set of lexicalized word structures by $\mathcal{W}$, and the set of unknown flat words by $\mathcal{F}$, then the joint probability $\Pr(T, S)$ according to our model is

$$\Pr(T, S) = \prod_{c \in \mathcal{C}} \Pr(c) \prod_{w \in \mathcal{W}} \Pr(w) \prod_{f \in \mathcal{F}} \Pr(f) \tag{10}$$

In practice, the logarithm of this probability can be calculated instead to avoid numerical difficulties.

5 The Parsing Algorithm

To find the parse tree with highest probability we use a chart parser adapted from Collins (1999). Two key changes must be made to the search process, though. Firstly, because we are proposing a new paradigm for Chinese word segmentation, the input to the parser must be raw sentences by definition. Hence to use the bottom-up parser, we need a lexicon of all characters together with what roles they can play in a flat word. We can get this lexicon from the treebank. For example, from the word 中央/NNf 'center', we can extract a role bNNf for character 中 'middle' and a role eNNf for character 央 'center'. The role bNNf means the beginning of the flat label NNf, while eNNf stands for the end of the label NNf. This scheme was first proposed by Luo (2003) in his character-based Chinese parser, and we find it quite adequate for our purpose here.

Secondly, in the bottom-up parser for head driven models, whenever a new edge is found, we must assign it a probability and a head word. If the newly discovered constituent is a flat word (its label ends with 'f'), then we set its head word to be the concatenation of all its child characters, i.e. the word itself. If it is an unknown word, we use (7) to assign the probability, otherwise its probability is set to be 1. On the other hand, if the new edge is a phrase or word with internal structures, the probability is set according to (2), while the head word is found with the appropriate head rules. In this bottom-up way, the probability for a complete parse tree is known as soon as it is completed. This probability includes both word generation probabilities and constituent probabilities.

6 Evaluation

For several reasons, it is a little tricky to evaluate the accuracy of our model for integrated morphological and syntactic parsing. First and foremost, we currently know of no other same effort in parsing the structures of Chinese words, and we have to annotate word structures by ourselves. Hence there is no baseline performance to compare with. Secondly, simply reporting the accuracy of labeled precision and recall is not very informative because our parser takes raw sentences as input, and its output includes a lot of easy cases like word segmentation and part-of-speech tagging results.

Despite these difficulties, we note that higher-level constituent parsing results are still somewhat comparable with previous performance in parsing Penn Chinese Treebank, because constituent parsing does not involve word structures directly. Having said that, it must be pointed out that the comparison is meaningful only in a limited sense, as in previous literatures on Chinese parsing, the input is always word segmented or even part-of-speech tagged. That is, the bracketing in our case is around characters instead of words. Another observation is we can still evaluate Chinese word segmentation and part-of-speech tagging accuracy, by reading off the corresponding result from parse trees. Again because we split the words with internal structures into their components, comparison with other systems should be viewed with that in mind.

Based on these discussions, we divide the labels of all constituents into three categories:

Phrase labels are the labels in Peen Chinese Treebank for nonterminal phrase structures, including NP, VP, PP, etc.

POS labels represent part-of-speech tags such as NN, VV, DEG, etc.

Flat labels are generated in our annotation for words with no interesting structures. Recall that they always end with an 'f' such as NNf, VVf and DEGf, etc.

With this classification, we report our parser's accuracy for phrase labels, which is approximately the accuracy of constituent parsing of Penn Chinese Treebank. We report our parser's word segmentation accuracy based on the flat labels. This accuracy is in fact the joint accuracy of segmentation and part-of-speech tagging. Most importantly, we can report our parser's accuracy in recovering word structures based on POS labels and flat labels, since word structures may contain only these two kinds of labels.

With the standard split of CTB 5.0 data into training, development and test sets (Zhang and Clark, 2009), the result are summarized in Table 1. For all label categories, the PARSEEVAL measures (Abney et al., 1991) are used in computing the labeled precision and recall.

Types	LP	LR	F_1
Phrase	79.3	80.1	79.7
Flat	93.2	93.8	93.5
Flat*	97.1	97.6	97.3
POS & Flat	92.7	93.2	92.9

Table 1: Labeled precision and recall for the three types of labels. The line labeled ‘Flat*’ is for unlabeled metrics of flat words, which is effectively the ordinary word segmentation accuracy.

Though not directly comparable, we can make some remarks to the accuracy of our model. For constituent parsing, the best result on CTB 5.0 is reported to be 78% F_1 measure for unlimited sentences with automatically assigned POS tags (Zhang and Clark, 2009). Our result for phrase labels is close to this accuracy. Besides, the result for flat labels compares favorably with the state of the art accuracy of about 93% F_1 for joint word segmentation and part-of-speech tagging (Jiang et al., 2008; Kruengkrai et al., 2009). For ordinary word segmentation, the best result is reported to be around 97% F_1 on CTB 5.0 (Kruengkrai et al., 2009), while our parser performs at 97.3%, though we should remember that the result concerns flat words only. Finally, we see the performance of word structure recovery is almost as good as the recognition of flat words. This means that parsing word structures accurately is possible with a generative model.

It is interesting to see how well the parser does in recognizing the structure of words that were not seen during training. For this, we sampled 100 such words including those with prefixes or suffixes and personal names. We found that for 82 of these words, our parser can correctly recognize their structures. This means our model has learnt something that generalizes well to unseen words.

In error analysis, we found that the parser tends to over generalize for prefix and suffix characters. For example, 大作家 ‘great writer’ is a noun phrase consisting of an adjective 大 ‘great’ and a noun 作家 ‘writer’, as shown in Figure 10(a), but our parser incorrectly analyzed it into a root 大作 ‘masterpiece’ and a suffix 家 ‘expert’, as in Figure 10(b). This is because the character 家 ‘expert’ is a very productive suffix, as in 化学家 ‘chemist’ and 外交家 ‘diplomat’. This observation is illuminating because most errors of our parser follow this pattern. Currently we don’t have any non-ad hoc way of preventing such kind of over generalization.

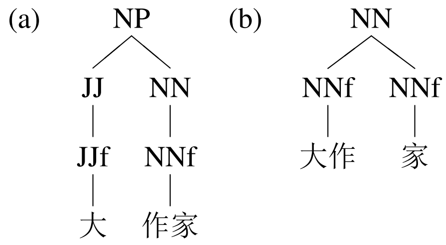

Figure 10: Example of parser error. Tree (a) is correct, and (b) is the wrong result by our parser.

7 Conclusion and Discussion

In this paper we proposed a new paradigm for Chinese word segmentation in which not only flat words were identified but words with structures were also parsed. We gave good reasons why this should be done, and we presented an effective method showing how this could be done. With the progress in statistical parsing technology and the development of large scale treebanks, the time has now come for this paradigm shift to happen. We believe such a new paradigm for word segmentation is linguistically justified and pragmatically beneficial to real world applications. We showed that word structures can be recovered with high precision, though there’s still much room for improvement, especially for higher level constituent parsing.

Our model is generative, but discriminative models such as maximum entropy technique (Berger et al., 1996) can be used in parsing word structures too. Many parsers using these techniques have been proved to be quite successful (Luo, 2003; Fung et al., 2004; Wang et al., 2006). Another possible direction is to combine generative models with discriminative reranking to enhance the accuracy (Collins and Koo, 2005; Charniak and Johnson, 2005).

Finally, we must note that the use of flat labels such as “NNf” is less than ideal. The most important reason these labels are used is we want to compare the performance of our parser with previous results in constituent parsing, part-of-speech tagging and word segmentation, as we did in Section 6. The problem with this approach is that word structures and phrase structures are then not treated in a truly unified way, and besides the 33 part-of-speech tags originally contained in Penn Chinese Treebank, another 33 tags ending with 'f' are introduced. We leave this problem open for now and plan to address it in future work.

Acknowledgments

I would like to thank Professor Maosong Sun for many helpful discussions on topics of Chinese morphological and syntactic analysis. The author is supported by NSFC under Grant No. 60873174. Heartfelt thanks also go to the reviewers for many pertinent comments which have greatly improved the presentation of this paper.

References

S. Abney, S. Flickenger, C. Gdaniec, C. Grishman, P. Harrison, D. Hindle, R. Ingria, F. Jelinek, J. Klavans, M. Liberman, M. Marcus, S. Roukos, B. Santorini, and T. Strzalkowski. 1991. Procedure for quantitatively comparing the syntactic coverage of English grammars. In E. Black, editor, *Proceedings of the workshop on Speech and Natural Language*, HLT '91, pages 306–311, Morristown, NJ, USA. Association for Computational Linguistics.

Adam L. Berger, Vincent J. Della Pietra, and Stephen A. Della Pietra. 1996. A maximum entropy approach to natural language processing. *Computational Linguistics*, 22(1):39–71.

Daniel M. Bikel and David Chiang. 2000. Two statistical parsing models applied to the Chinese treebank. In *Second Chinese Language Processing Workshop*, pages 1–6, Hong Kong, China, October. Association for Computational Linguistics.

Eugene Charniak and Mark Johnson. 2005. Coarse-to-fine n-best parsing and maxent discriminative reranking. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 173–180, Morristown, NJ, USA. Association for Computational Linguistics.

Eugene Charniak. 1997. Statistical parsing with a context-free grammar and word statistics. In *Proceedings of the fourteenth national conference on artificial intelligence and ninth conference on Innovative applications of artificial intelligence*, AAAI'97/IAAI'97, pages 598–603. AAAI Press.

Michael Collins and Terry Koo. 2005. Discriminative reranking for natural language parsing. *Computational Linguistics*, 31:25–70, March.

Michael Collins. 1999. *Head-Driven Statistical Models for Natural Language Parsing*. Ph.D. thesis, University of Pennsylvania.

Michael Collins. 2003. Head-driven statistical models for natural language parsing. *Computational Linguistics*, 29(4):589–637.

Pascale Fung, Grace Ngai, Yongsheng Yang, and Benfeng Chen. 2004. A maximum-entropy Chinese parser augmented by transformation-based learning. *ACM Transactions on Asian Language Information Processing*, 3:159–168, June.

Jianfeng Gao, Andi Wu, Cheng-Ning Huang, Hong qiao Li, Xinsong Xia, and Hauwei Qin. 2004. Adaptive Chinese word segmentation. In *Proceedings of the 42nd Meeting of the Association for Computational Linguistics (ACL'04), Main Volume*, pages 462–469, Barcelona, Spain, July.

Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2005. Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics*, 31(4):531–574.

Yoav Goldberg and Reut Tsarfaty. 2008. A single generative model for joint morphological segmentation and syntactic parsing. In *Proceedings of ACL-08: HLT*, pages 371–379, Columbus, Ohio, June. Association for Computational Linguistics.

Wenbin Jiang, Liang Huang, Qun Liu, and Yajuan Lü. 2008. A cascaded linear model for joint Chinese word segmentation and part-of-speech tagging. In *Proceedings of ACL-08: HLT*, pages 897–904, Columbus, Ohio, June. Association for Computational Linguistics.

Wenbin Jiang, Liang Huang, and Qun Liu. 2009. Automatic adaptation of annotation standards: Chinese word segmentation and POS tagging – a case study. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 522–530, Suntec, Singapore, August. Association for Computational Linguistics.

Canasai Kruengkrai, Kiyotaka Uchimoto, Jun'ichi Kazama, Yiou Wang, Kentaro Torisawa, and Hitoshi Isahara. 2009. An error-driven word-character hybrid model for joint Chinese word segmentation and POS tagging. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Process-*

ing of the AFNLP, pages 513–521, Suntec, Singapore, August. Association for Computational Linguistics.

Zhongguo Li and Maosong Sun. 2009. Punctuation as implicit annotations for Chinese word segmentation. *Computational Linguistics*, 35:505–512, December.

Junhui Li, Guodong Zhou, and Hwee Tou Ng. 2010. Joint syntactic and semantic parsing of Chinese. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 1108–1117, Uppsala, Sweden, July. Association for Computational Linguistics.

Xiaoqiang Luo. 2003. A maximum entropy Chinese character-based parser. In Michael Collins and Mark Steedman, editors, *Proceedings of the 2003 Conference on Empirical Methods in Natural Language Processing*, pages 192–199.

Hwee Tou Ng and Jin Kiat Low. 2004. Chinese part-of-speech tagging: One-at-a-time or all-at-once? word-based or character-based? In Dekang Lin and Dekai Wu, editors, *Proceedings of EMNLP 2004*, pages 277–284, Barcelona, Spain, July. Association for Computational Linguistics.

Richard Sproat and Thomas Emerson. 2003. The first international Chinese word segmentation bakeoff. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 133–143, Sapporo, Japan, July. Association for Computational Linguistics.

Richard Sproat, William Gale, Chilin Shih, and Nancy Chang. 1996. A stochastic finite-state word-segmentation algorithm for Chinese. *Computational Linguistics*, 22(3):377–404.

Honglin Sun and Daniel Jurafsky. 2004. Shallow semantc parsing of Chinese. In Daniel Marcu Susan Dumais and Salim Roukos, editors, *HLT-NAACL 2004: Main Proceedings*, pages 249–256, Boston, Massachusetts, USA, May 2 - May 7. Association for Computational Linguistics.

Mengqiu Wang, Kenji Sagae, and Teruko Mitamura. 2006. A fast, accurate deterministic parser for chinese. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 425–432, Sydney, Australia, July. Association for Computational Linguistics.

Andi Wu and Jiang Zixin. 1998. Word segmentation in sentence analysis. In *Proceedings of the 1998 International Conference on Chinese information processing*, Beijing, China.

Andi Wu. 2003. Customizable segmentation of morphologically derived words in Chinese. *Computational Linguistics and Chinese language processing*, 8(1):1–28.

Nianwen Xue, Fei Xia, Fu-Dong Chiou, and Martha Palmer. 2005. The Penn Chinese Treebank: phrase structure annotation of a large corpus. *Natural Language Engineering*, 11(2):207–238.

Nianwen Xue. 2003. Chinese word segmentation as character tagging. *Computational Linguistics and Chinese Language Processing*, 8(1):29–48.

Yue Zhang and Stephen Clark. 2007. Chinese segmentation with a word-based perceptron algorithm. In *Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics*, pages 840–847, Prague, Czech Republic, June. Association for Computational Linguistics.

Yue Zhang and Stephen Clark. 2008. Joint word segmentation and POS tagging using a single perceptron. In *Proceedings of ACL-08: HLT*, pages 888–896, Columbus, Ohio, June. Association for Computational Linguistics.

Yue Zhang and Stephen Clark. 2009. Transition-based parsing of the Chinese treebank using a global discriminative model. In *Proceedings of the 11th International Conference on Parsing Technologies*, IWPT '09, pages 162–171, Morristown, NJ, USA. Association for Computational Linguistics.

Hai Zhao. 2009. Character-level dependencies in Chinese: Usefulness and learning. In *Proceedings of the 12th Conference of the European Chapter of the ACL (EACL 2009)*, pages 879–887, Athens, Greece, March. Association for Computational Linguistics.

Guodong Zhou and Jian Su. 2003. A Chinese efficient analyser integrating word segmentation, part-of-speech tagging, partial parsing and full parsing. In *Proceedings of the Second SIGHAN Workshop on Chinese Language Processing*, pages 78–83, Sapporo, Japan, July. Association for Computational Linguistics.