Applying Maximum Entropy to Robust Chinese Shallow Parsing

Shih-Hung Wu*†, Cheng-Wei Shih†, Chia-Wei Wu†, Tzong-Han Tsai†,
and Wen-Lian Hsu†

†*Institute of Information Science, Academia Sinica, Taiwan, R.O.C*
(shwu,dapi,cwwu,thtsai,hsu)@iis.sinica.edu.tw

**Dep. Of CSIE, Chaoyang University of Technology, Taichung County, Taiwan, R.O.C*
shwu@cyut.edu.tw

Abstract

Recently, shallow parsing has been applied to various information processing systems, such as information retrieval, information extraction, question answering, and automatic document summarization. A shallow parser is suitable for online applications, because it is much more efficient and less demanding than a full parser. In this research, we formulate shallow parsing as a sequential tagging problem and use a supervised machine learning technique, Maximum Entropy (ME), to build a Chinese shallow parser. The major features of the ME-based shallow parser are POSs and the context words in a sentence. We adopt the shallow parsing results of Sinica Treebank as our standard, and select 30,000 and 10,000 sentences from Sinica Treebank as the training set and test set respectively. We then test the robustness of the shallow parser with noisy data. The experiment results show that the proposed shallow parser is quite robust for sentences with unknown proper nouns.

1. Introduction

Parsing is a basic technique in natural language processing; however, a full parser is usually costly and slow. Recently, shallow parsing has been applied to various information processing systems [12]. Compared to the performance of full parsers, a shallow parser is much faster and the parsing result is more useful for various applications, such as information retrieval and extraction, question answering, and automatic document summarization. In this paper, we adopt a machine learning approach to the Chinese shallow parsing problem.

Chinese full parsing is very challenging,[18, 22] because it is difficult to achieve high accuracy, and the performance is not suitable for online applications. Shallow parsing of Chinese, on the other hand, is promising and desirable in terms of efficiency. Researchers in Beijing, Harbin, Shenyang, and Hong Kong have also developed related techniques [10, 15, 16, 20, 21]. Most of these works use machine learning approaches, instead of the rule-based approach used in full parsing. Popular machine learning methods such as SVM, CRF, and ME, have been tested. The parsing speed of each approach is fast and the parsing accuracy is acceptable.

Currently, there is no standard for Chinese shallow parsing. Li et. al. [9] developed a Chinese shallow parsed treebank to extract Chinese collocations automatically and built a large collocation bank. There are also some works on a standard for Chinese shallow parsing [9, 19, 20]. Nevertheless, the POS standard and vocabulary in each approach are different; thus, between simplified Chinese and traditional Chinese, we cannot adopt their standard for simplified Chinese to traditional Chinese. Instead, we use the first level of the parsing results of Sinica Treebank as our shallow parsing standard [4]. Originally, Sinica Treebank was designed to provide full parsing results, whereby sentences could be labeled with POS tags and the full parsing structure. There are 54,000 sentences in Sinica Treebank, from which we randomly selected 30,000 and 10,000 sentences as the training set and test set respectively.

Since there are many unknown words in Chinese [11], a Chinese shallow parser must be robust against such words [22]. For example, it is not hard to correctly chunk the sentence “高漸離/擊筑/的/音調/忽然/急轉成/悲壯” into “高漸離擊筑的音調/NP 忽然/Dd 急轉成/DM 悲壯/VP”, if we know that “高漸離” is a proper noun. However, if the name is unknown, it could be split into three single characters and tagged with the three POS of the single characters, i.e., “高/漸/離 [VH13/Dd/P15]”. It might then be incorrectly chunked as ”高漸/NP 離擊筑的音調/PP 忽然/Dd 急轉成/DM 悲壯/VP”. In this research, we simulate unknown words by adding some noises to the corpus in order to test the robustness of the shallow parser. Since new proper nouns are normally unknown, we design three ways to add noises to the training and testing sets by treating proper nouns as unknown words.

2. Shallow Parsing Standard

Sinica Treebank provides a full parse tree for each sentence. Here, we use the first-layer parsing results of Sinica Treebank as the standard for shallow parsing. Instead of using all the phrase tags in Sinica Treebank, we annotate five of them for chunking; all other phrases (including single words not in any phrase) are tagged as others (X). The five tags, namely, noun phrase (NP), verb phrase (VP), preposition phrase (PP), geographic phrase (GP), and clause (S), are the major tags in Sinica Treebank, and therefore play significant syntactical roles. Thus, the constituents of the root node of a parse tree are NP, VP, PP, GP, S, and X. Table 1 lists examples of the six types of constituent.

Table 1. Chunk Tags

Chunk Tag	Description	Example
NP	Noun Phrase	前十名 / 的 / 選手 [DM / DE / Nab]
VP	Verb Phrase	傳遞 / 區運 / 聖火 [VD1 / Nad / Nac]
PP	Preposition Phrase	在 / 旅客 / 口 / 中 [P21 / Nab / Nab / Ncda]
GP	Geographic Phrase	一個 / 星期 / 以後 [DM / Nac / Ng]
S	Clause	窗戶 / 玻璃 / 破掉 [Nab / Nab / VH11]
X	Others	０ / 到 / ２度 [DM / Caa / DM]

3. A Maximum Entropy-based Shallow Parser

Parsing is a fundamental technique in natural language processing, the results of which can be used to improve various natural language tasks, such as word-sense disambiguation (WSD) [3] and part-of-speech (POS) tagging [12].

Many natural language processing tasks, such as part-of-speech tagging, named-entity recognition, and shallow parsing, can be viewed as sequence analysis tasks. Shallow parsing identifies the non-recursive core of each phrase type in a text as a precursor to full parsing or information extraction [1, 6]. The paradigmatic shallow parsing problem is called NP chunking, which finds the non-recursive cores of noun phrases called base NPs. Ramshaw and Marcus introduced NP chunking as a machine-learning problem [14].

Machine learning techniques, such as maximum entropy (ME) and conditional random fields (CRF), are quite popular for sequential tagging. We adopt ME to build a robust Chinese shallow parser.

3.1 The B-I-O Scheme of Our Shallow Parser

In this work, we regard each word as a token, and consider a test corpus and a set of *n* phrase categories. Since a phrase can have more than one token, we associate two tags, *x*: *x_begin* and *x_continue*, with each category. In addition, we use the tag *others* to indicate that a token is not part of a phrase. The shallow parsing problem can then be redefined as a problem of assigning one of $2n + 1$ tags to each token. This is called the B-I-O scheme. There are 5 named entity categories and 11 tags: *NP_begin, NP_continue, VP_begin, VP_continue, PP_begin, PP_continue, GP_begin, GP_continue, S_begin, S_continue,* and *X(others).*

3.2 Maximum Entropy Formula

ME is a flexible statistical model that assigns an *outcome* to each token based on its *history* and *features* [2]. The outcome space is comprised of the tags for an ME formulation. ME computes the probability $p(o|h)$ for any o from the space of all possible outcomes, O, and for every h from the space of all possible histories, H. A *history* is composed of all the conditioning data that enables one to assign probabilities to the space of outcomes. In shallow parsing, *history* can be viewed as all the information derived from the test corpus relevant to the current token.

The computation of $p(o|h)$ in ME depends on a set of binary-valued *features*, which is helpful in making a prediction about the outcome. For instance, one of our features is as follows: when the current token is a verb, it is likely to be the leading character of a verb phrase. More formally, we can represent this feature as

$$f(h,o) = \begin{cases} 1: \text{if Current - token - verb(h)} = \text{true and } o = VP\_begin \\ 0: \text{else} \end{cases} \quad (1)$$

Here, *Current-token-verb(h)* is a binary function that returns the value *true* if the *current token* of the history *h* is a verb.

Given a set of features and a training corpus, the ME estimation process produces a model in

which every feature f_i has a weight α_i. This allows us to compute the conditional probability as follows:

$$p(o \mid h) = \frac{1}{Z(h)} \prod_i \alpha_i^{f_i(h,o)}, \qquad (2)$$

where Z(h) is a normalization factor. Intuitively, the probability is the multiplication of the weights of active features (i.e., those f_i (h,o) = 1). The weight α_i is estimated by means of a procedure called Generalized Iterative Scaling (GIS) [8], which improves the estimation of the weights at each iteration. The ME estimation technique guarantees that, for every feature f_i, the expected value of α_i will be equal to the empirical expectation of α_i in the training corpus. ME allows the designer to concentrate on finding the features that characterize the problem, while letting the ME estimation routine deal with assigning relative weights to the features.

3.3 Decoding

After an ME model has been trained and the proper weight α_i has been assigned to each feature f_i, decoding (i.e., *marking up*) a new piece of text becomes a simple task. First, the model tokenizes the text and preprocesses the test sentence. Then, for each token, it checks which features are active and combines the α_i of the active features according to Equation 2. Finally, a Viterbi search is run to find the highest probability path through the lattice of conditional probabilities that does not produce any invalid tag sequences. Further details of the Viterbi search can be found in [17].

4. Experiment

By comparing models with and without noisy training data, we can determine whether our Chinese shallow parser is noisy-data-tolerant. In this section, we describe how we add noisy data to maximum entropy models and evaluate the tolerance of our system to Chinese chunking.

4.1 Data and Features

Sinica Treebank contains more than 54,000 sentences, from which we randomly extract 30,000 for training and 10,000 for testing. The tokenized results and the corresponding part-of-speech sequences of these sentences are extracted into a feature file, and the top-level chunks of the parsing tree structure can be taken as the standard for training and evaluation. The information in the feature file is translated into machine learning features by ME model in both the training and testing phrases. The features we adopted are: words, adjacent characters, prefixes of words (1 and 2 characters), suffixes of words (1 and 2 characters), word length, POS of words, adjacent POS tags, and the word's location in the chunk it belongs to.

To analyze the performance of our shallow parser under noisy conditions, we build a standard model and various noisy models. Training data consisting of the tokenization and POS information derived from the manually annotated Sinica Treebank is used as the standard model in our experiments. The accuracy of chunking in this model is then compared with that of models containing noise to

observe the difference.

4.2 Noise Model Generation

The most important issue in noisy model generation is how to mix noisy features with correct features as smoothly as in a real parsing system. We design three methods for adding noise to generate different types of models with noisy tokenization and POS sequences.

The first two approaches are based on unknown word replacement. We find that unknown words are one of the major causes of noisy data in real world system processing, because most unknown words are proper nouns. Theoretically, we can pick a certain number of proper nouns in the selected data and substitute them with noisy data to simulate real world input. In our experiment, “Nb” and “Nc”, which are defined as “proper nouns” and “proper location nouns” respectively in the Sinica Treebank tagging guideline [5], are chosen as replacement targets. Words with these two target POS are regarded as replacement target strings and replaced by noisy data.

We adopt two types of noisy data for unknown word replacement. The first is the split character sequence of a replacement target string in a sentence. Initially, we extract the correct tokenization results and POS sequences of all data in the Sinica Treebank with “Nb” and “Nc”. Then, wherever applicable, we split the replacement target string in a sentence into single Chinese characters. The corresponding POS tag of each split character is re-assigned by selecting the most frequent POS tags of these single characters in Sinica Treebank. For example, “馬來西亞” (Malaysia) would be split into “馬”, ”來”, ”西”, and “亞”, and the original POS tag “Nca” would be replaced by the pos tags of four single characters: “Nab”, “Dbab”, “Ncda”, and “Nca”. In this experiment, we control the amount of noisy data in models to observe the relation between the percentage of imprecise data and the chunking performance. The model generated by this approach is called a Type 1 noise model. Another approach, called the Type 2 noise model, tokenizes the replacement target with AUTOTAG, which may produce segmenting boundaries and POS tags that differ from those in Sinica Treebank. The information is then used as noisy features and replaces the target string. For instance, the replacement target string “太白金星” with POS tag ”Nb” would be tagged by AUTOTAG as “太白/Nb” and “金星/Nb”. The above noise-adding approaches are used to generate training data, as well as various kinds of noisy information in the test sets.

In addition, we adopt an automatic tool, CKIP AUTOTAG [7], to obtain the tokenization information and POS features for generating models. This is a Chinese tokenizing tool that can deal with word segmentation in both the training and testing sets. CKIP AUTOTAG provides the POS sequences of the sentences. The tokenized sentences and POS sequences produced by AUTOTAG are used to generate feature files for ME processing.

5. Results and Discussion

In our experiment, we adopt the B-I-O scheme to identify the boundaries of Chinese chunks and the position of each element word in the chunks. In addition, we employ the following four standards

when calculating the accuracy of Chinese shallow parsing: evaluation by token, by chunk boundary, by chunk category sequence, and by chunks. Token evaluation is based on the number of Chinese words. All words in the test data can be verified independently to determine if they have the correct boundaries and belong to the right chunks. Evaluation by chunk boundary only checks the boundaries of each chunk, while evaluation by chunk category sequence only checks if all the chunks in a sentence can be identified successfully and disregards the constituents. By contrast, in chunk evaluation, the basic unit is the whole chunk, and only a chunk with the right constituents and tagged with proper categories can be considered correct. We use an example to demonstrate the evaluation process. The input sentence is "小朋友 換成 你 來 試試看", which consists of five tokens; and the standard parsing result is "小朋友/NP 換成/VC 你/NP 來-試試看/VP", which contains four chunks. The parsing result we obtain from the system is "小朋友/NP 換成/VC 你/NP 來/Db 試試看/VE", which contains five chunks. In this case, the accuracy of the chunk boundary and the chunk category are both 3/4=0.75, because the first three chunks in the sentence have the correct boundaries and phrase tags, and the last VP chunk is separated by two units. The token number in this sentence is 5 and the last two tokens have incorrect phrase category tags. Therefore, the accuracy of the token is 3/5=0.6. In chunk evaluation, three of the four chunks are identified successfully and the chunk accuracy is 3/4=0.75. We adopt these evaluation methods in all the experiment configurations in Tables 2 to 5.

5.1 Performance on Noisy Data

Table 2 shows the accuracy rates using Type 1 noisy models with different scales of noisy data for chunking clean test data. The columns show the percentage of 'Nb' and 'Nc' replaced by single character noisy data in the training model, and the rows indicate the four evaluation methods. We find that the accuracy in this series decreases slightly, while the percentage of single character noisy data increases.

Table 2. Results of chunking clean test data with the Type 1 noise model

	Boundary	Category	Tokens	Chunks
0 (%)	84.83	70.10	69.14	70.47
10 (%)	84.74	69.92	69.04	70.30
20 (%)	84.80	69.94	69.03	70.26
30 (%)	84.77	69.88	69.10	70.20
40 (%)	84.70	69.77	68.97	70.13
50 (%)	84.64	69.65	69.02	70.00
60 (%)	84.56	69.57	68.78	69.82
70 (%)	84.42	69.39	68.76	69.59
80 (%)	84.53	69.67	68.99	69.77
90 (%)	84.38	69.44	68.58	69.72
100 (%)	84.26	69.51	68.57	69.75

Table 3 shows the accuracy rates using the Type 1 model with different scales of noisy data for chunking test data with single character noise (Type 1). It is quite interesting that the curve is not monotonically increasing or decreasing. This indicates that the accuracy in this series decreases until the percentage of noise reaches 60%, and then it increases. Figures 1 to 4 show the differences between the clean test data and the noisy test data in Tables 2 and 3. We can observe the trends in the experiment results more intuitively.

Table 3. Results of chunking test data containing Type 1 noisy data with the Type 1 noise model

	Boundary	Category	Tokens	Chunks
0 (%)	83.73	69.09	65.16	66.51
10 (%)	83.65	69.00	65.33	66.36
20 (%)	83.72	69.11	65.34	66.30
30 (%)	83.69	69.20	65.37	66.27
40 (%)	83.62	69.14	65.42	66.25
50 (%)	83.58	69.05	65.52	66.13
60 (%)	83.57	69.07	65.36	66.00
70 (%)	83.52	69.24	65.70	66.07
80 (%)	83.63	69.46	65.83	66.25
90 (%)	83.65	69.49	65.69	69.30
100 (%)	83.77	69.67	65.85	66.42

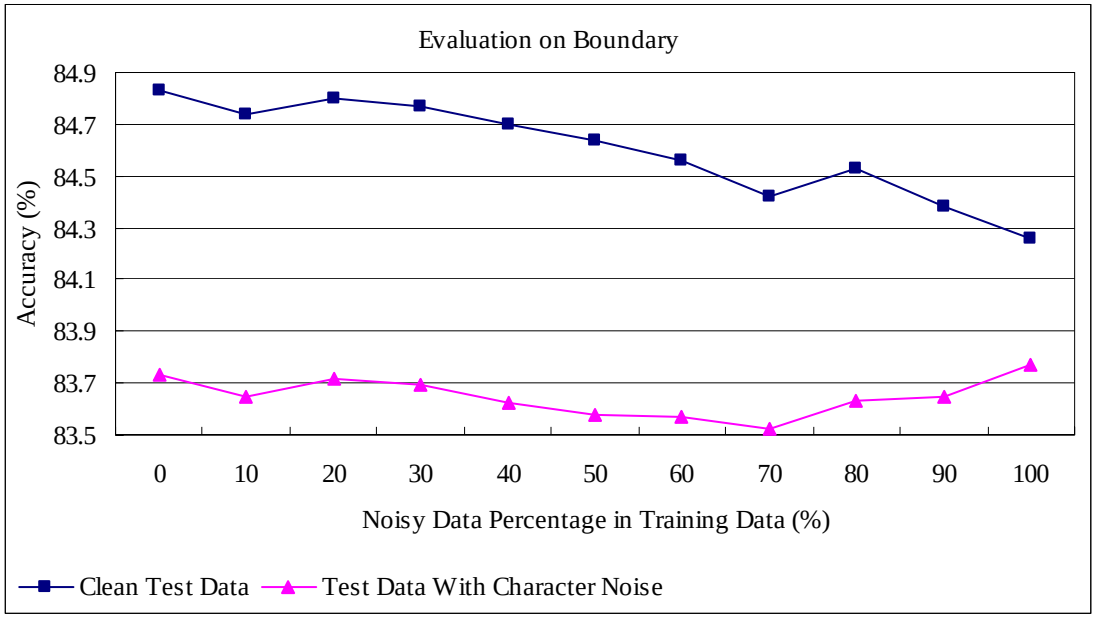

Figure 1. Evaluation of the boundaries in different experiment configurations

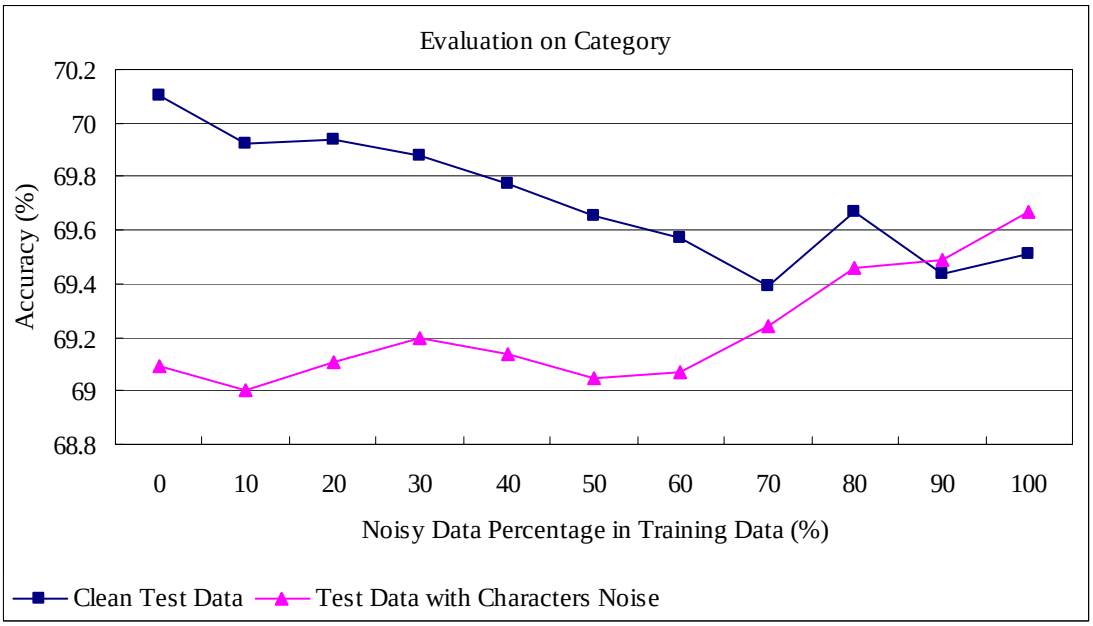

Figure 2. Evaluation of the chunking category in different experiment configurations

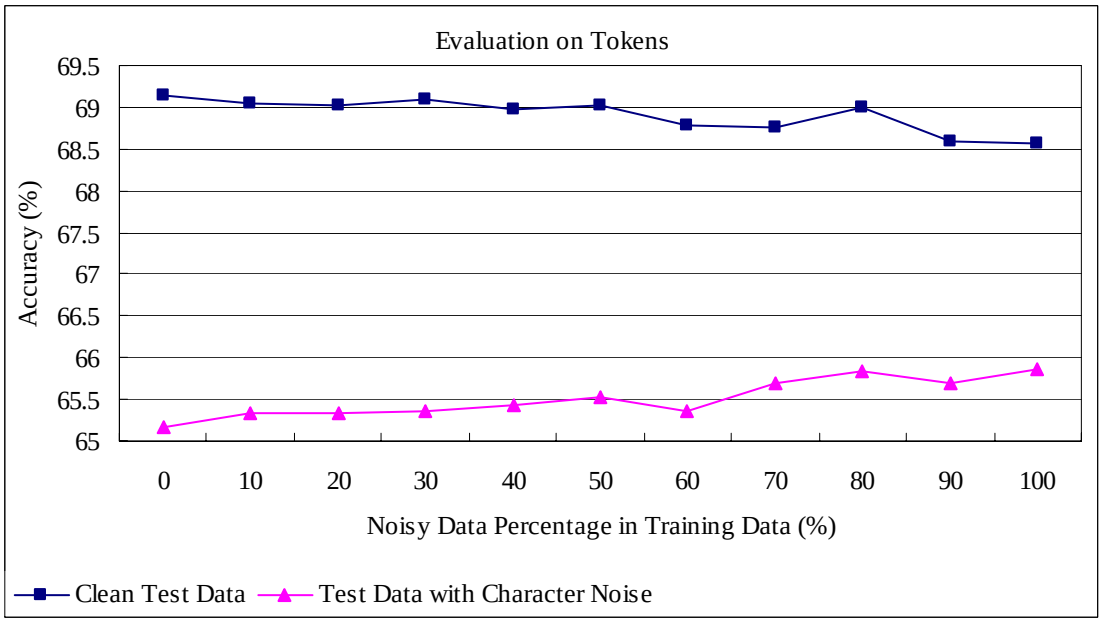

Figure 3. Evaluation of tokens in different experiment configurations

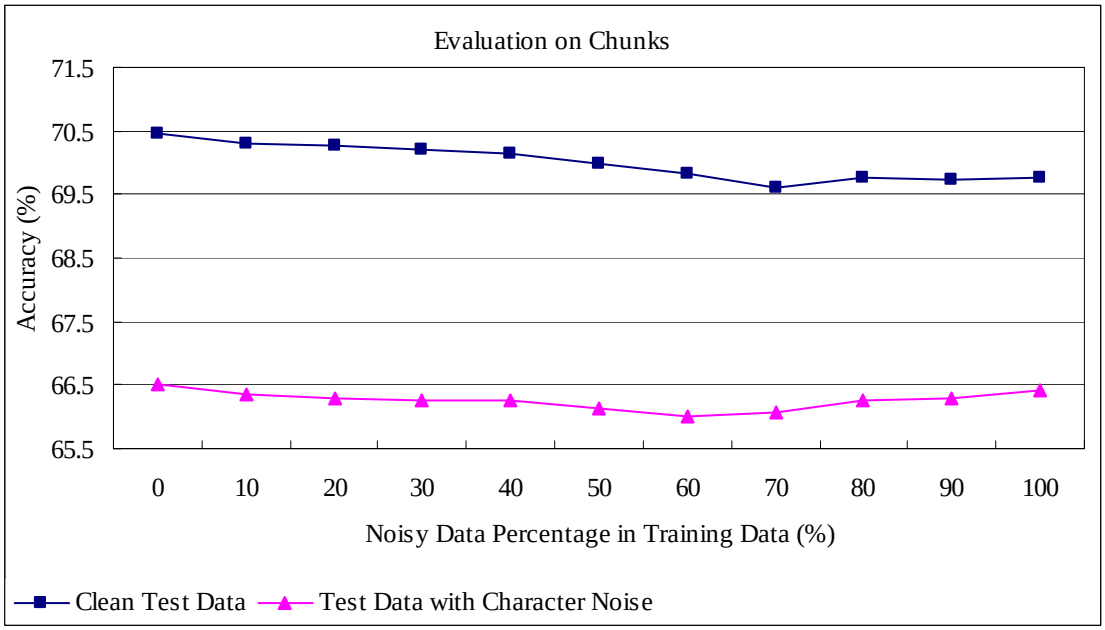

Figure 4. Evaluation of chunks in different experiment configurations

Table 4 shows the accuracy rates using the Type 2 noise model with and without tokenized strings for chunking clean test sentences and test data with tokenized strings. There are four configurations:

- C-C: Using a clean training model and clean test data.
- C-N: Using a clean training model and noisy test data in which all 'Nb' and 'Nc' are replaced by tokenized results.
- N-C: Using a training model with noisy data in which all 'Nb' and 'Nc' are replaced by the tokenized results of chunking clean test data.
- N-N: Both the training model and the test data have noisy data in which all 'Nb' and 'Nc' are replaced by tokenized results.

Table 4 also shows that noisy training data yields better accuracy for both clean and noisy test data, although the difference is quite small.

Table 4. Results of chunking with the Type 2 noise model

	Boundary	Category	Tokens	Chunks
C-C	84.83	70.10	69.14	70.47
C-N	84.84	70.09	69.04	70.37
N-C	84.89	70.13	69.15	70.51
N-N	84.90	70.11	69.02	70.38

Table 5 shows the accuracy rates using the model generated by AUTOTAG-parsed data and Sinica Treebank chunking tags. Both the training and the test sets are preprocessed by AUTOTAG. This experiment is designed for open testing; thus, we can use the AUTOTAG program to tokenize any

sentence and give it POS tags. However, compared to the standard model, the chunking accuracy is lower. The parsing results of the AUTOTAG-parsed model and the Type 2 noise models are shown in Figure 5.

Table 5. Accuracy using the model generated by AUTOTAG-parsed data

	Boundary	Category	Tokens	Chunks
Fully AUTOTAG	81.42	64.81	61.80	61.30

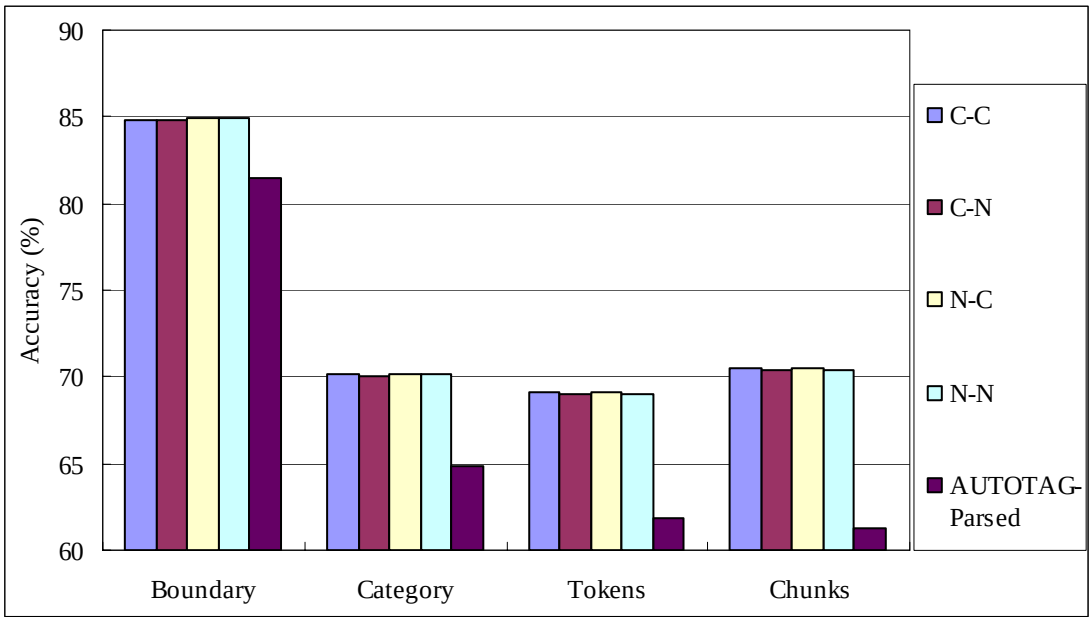

Figure 5. Comparison of various experiment configurations using tokenized string noisy data (the Type 2 noise model) and the AUTOTAG-parsed model

In Tables 6, 7, and 8, we give examples of the correct and incorrect shallow parsing results of four sentences. In each table, the left column contains the original sentences tokenized and tagged with POS tags; the center column shows the standard chunking result from Sinica Treebank; and the right column shows the shallow parsing result of our system. Table 6 shows the parsing examples with Type 1 noise. The shallow parsing results of the first two sentences are correct, while those of the last two sentences are incorrect.

Table 6. Shallow parsing examples with Type 1 noise

Sentence and POS sequences with Type 1 noise	Chunking standard from Sinica Treebank	Chunking results of our system
女性/形象/在/**台/灣**/和/**中/國/大/陸**/小說/是/解放/的/過程 [Nab/Nac/P21/**Nca/Nab**/Caa/**Ng/Ncb/VH13/Nab**/Nac/V_11/VC2/DE/Nac]	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP

與/**中/華**/及/**日/本/隊**/在/伯仲/之間 [P35/**Ng/Nca**/Caa/**Nca/Nes/Nab**/VC1/Nhac/Ng]	與中華及日本隊/PP 在/VC 伯仲之間/GP	與/P3 中華及日本隊/NP 在/VC 伯仲之間/GP
首先/義賣/的/是/**黑/將/軍/史/東**/的/手套 [Cbbb/VC31/DE/V_11/**VH11/Dd/Nab/Nad/Ncda**/DE/Nab]	首先義賣的/NP 是/V_ 黑將軍史東的手套/NP	首先/Cb 義賣的/NP 是/V_ 黑將軍史東的手套/NP *
學藝/股長/**王/文/星**/站起來/說 [Nad/Nab/**Nbc/Nab/Nab**/VA11/VE2]	學藝股長王文星/NP 站起來/VA 說/VP	學藝股長王文星站起來/NP 說/VP *

Table 7 shows the parsing examples with Type 2 noise. The shallow parsing results of the first and the last sentences are correct, while those of the second and the third sentences are incorrect.

Table 7. Shallow parsing examples with Type 2 noise

Sentence and POS sequences with Type 2 noise	Chunking standard from Sinica Treebank	Chunking results of our system
女性/形象/在/台灣/和/**中國/大陸**/小說/是/解放/的/過程 [Nab/Nac/P21/Nca/Caa/**Nc/Nc**/Nac/V_11/VC2/DE/Nac]	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP
與/中華/及/**日本/隊**/在/伯仲/之間 [P35/Nba/Caa/**Nc/Na**/VC1/Nhac/Ng]	與中華及日本隊/PP 在/VC 伯仲之間/GP	與/P3 中華及日本隊/NP 在/VC 伯仲之間/GP *
首先/義賣/的/是/**黑/將軍**/史東/的/手套 [Cbbb/VC31/DE/V_11/**VH/Na**/Nba/DE/Nab]	首先義賣的/NP 是/V_ 黑將軍史東的手套/NP	首先/Cb 義賣的/NP 是/V_ 黑將軍史東的手套/NP *
學藝/股長/**王/文星**/站起來/說 [Nad/Nab/**Nb/Nb**/VA11/VE2]	學藝股長王文星/NP 站起來/VA 說/VP	學藝股長王文星/NP 站起來/VA 說/VP

Table 8 shows the parsing results using AUTOTAG-parsed training data and test data. The results of the first and last sentences are correct, while those of the second and the third sentences are incorrect. We replace the original word segmentation and POS tags of all the sentences with AUTOTAG-parsed word segmentation and POS tags. The word segmentation of the last sentence provided by AUTOTAG is incorrect; however, the chunking result is correct.

Table 8. Shallow parsing examples with AUTOTAG-parsed training data and test data

AUTOTAG-parsed Sentence and POS sequences	Chunking standard from Sinica Treebank	Chunking results of our system

女性/形象/在/台灣/和/中國/大陸/小說/是/解放/的/過程 [Na/Na/P/Nc/Caa/Nc/Nc/Na/SHI/VC/DE/Na]	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP	女性形象/NP 在台灣和中國大陸小說/PP 是/V_ 解放的過程/NP
與/中華/及/日本/隊/在/伯仲/之間 [P/Nc/Caa/Nc/Na/P/Nh/Ng]	與中華及日本隊/PP 在/VC 伯仲之間/GP	與中華及日本隊/PP 在伯仲之間/PP *
首先/義賣/的/是/黑/將軍/史東/的/手套 [D/VC/DE/SHI/VH/Na/Nb/DE/Na]	首先義賣的/NP 是/V_ 黑將軍史東的手套/NP	首先/Cb 義賣的/NP 是/V_ 黑將軍史東的手套/NP *
學藝/股長王/文星/站起來/說 [Na/Nb/Nb/VA/VE]	學藝股長王文星/NP 站起來/VA 說/VP	學藝股長王文星/NP 站起來/VA 說/VP

The experiment results show the noise-tolerance of our Chinese shallow parser with two different kinds of noise from unknown proper nouns. The system's performance is only degraded slightly when noisy data is added. Most sentences, such as "六十年代的台灣是怎樣的形貌" in which "台灣" is split into two characters and assigned with incorrect POS tags, can still be identified. However, the token accuracy is a little lower than the chunk accuracy, which indicates that our system needs to be improved for chunking longer phrases. In contrast, the chunking accuracy obviously decreases if models fully generated by AUTOTAG-parsed data are used. The difference between the AUTOTAG and Sinica Treebank tag sets probably causes the accuracy to decrease. Furthermore, this suggests that, while the shallow parsing system can deal with unknown nouns, it has difficulty dealing with other kinds of noisy data. For example, data preprocessing errors, such as, incorrect tokenization or wrong tagging in other POS categories, affect the performance of shallow parsing substantially We can not comment on which part-of-speech tags are the major factors in Chinese chunking without conducting additional experiments.

5.2 Use of Our Shallow Parser on News Articles

For the first application of our shallow parser, we collect some news articles as the test set. The articles did not have standard word segmentation, POS tagging, and parsing results; therefore, we cannot report on the accuracy. However, we find the results interesting. Some examples are given in Table 9. The left column shows the original sentences tokenized and tagged with POS tags by AUTOTAG. The right column shows the shallow parsing results using our system.

One interesting point is that the shallow parser tends to group named entities into a phrase. Therefore, the shallow parsing result can be used as a feature for boundary detection in named entity recognition (NER). In sentence 1, "中鋼公司" is grouped as one phrase, and in sentence 9, "中鋼公司 88 年盈餘" is grouped as one phrase, without first recognizing that "中鋼公司" is an entity by NER. Another example, in sentence 2 is that "益華在花蓮的三棟大樓" is grouped as one phrase, without first recognizing that "益華" is a company name.

Table 9. Shallow parsing results for news articles

	Tokenization and POS of Sentences	Shallow Parsing Result
1	中 鋼 / 公 司 / 是 / 台 灣 / 鋼 鐵 業 / 龍 頭 [Nc/Nc/SHI/Nc/Na/Na]	中鋼公司/NP 是 台灣鋼鐵業龍頭 /NP
2	益/華/在/花蓮/的/三棟/大樓/有/二/棟/是/七層/建築 [VJ/ Nc/P/Nc/DE/Nb/Na/V_2/Neu/Nf/SHI/Na/Na]	益華在花蓮的三棟大樓/NP 有 二棟/NP 是 七層建築/NP
3	大陸/仍/有/廣闊/發展/空間 [Nc/D/V_2/VH/VC/Na]	大陸/NP 仍 有 廣闊發展空間/NP
4	光/是/中共/國家/主席/江澤民/就/出/訪五次 [Da/SHI/Nb/Na/Na/Nb/D/VC/Na]	光/NP 是 中共國家主席江澤民/NP 就出訪五次/PP
5	許 多 / 地 區 / 都 / 出 現 / 新 舊 / 共 存 / 的 / 景 觀 [Neqa/Nc/D/VH/Na/VH/DE/Na]	許多地區/NP 都 出現 新舊共存的 景觀/NP
6	過去/一年/是/兩岸/關係/比較/困難/、/且/希望/落空/的/一年 [Nd/Nd/SHI/Nc/Na/Dfa/VH/PAUSECATEGORY/Cbb /VK/VH/DE/Nd]	過去一年/NP 是 兩岸關係比較困難、且希望落空的一年/NP
7	恆生/指數/創下/歷史/新高 [Nb/Na/VC/Na/VH]	恆生 指數/NP 創下 歷史新高/NP
8	將/資本主義/及/投機/氣息/帶入/大陸/內部 [P/Na/Caa/VH /Na/VCL/Nc/Ncd]	將資本主義及投機氣息/PP 帶入 大陸內部/NP
9	中鋼/公司/88/年盈餘/可望/達到/140 億/元/左右 [Nc/Nc/Neu/Na/VK/VJ/Neu/Nf/Ng]	中鋼公司 88 年盈餘/NP 可望達到 140 億元/VP 左右
10	企業界/已/開始/尾牙/聚餐 [Nc/D/VL/Nd/VA]	企業界/NP 已 開始 尾牙聚餐/VP
11	投資/人/靜候/美國/聯邦/準備/理事會/(/Fed/)/21 日/的/利率/決策 [Nc/Na/VJ/Nc/Na/VC/Na/PARENTHESISCATEGOR Y/FW /PARENTHESISCATEGORY/Nd/DE/Na/Na]	投資人/NP 靜候 美國聯邦準備理事會(Fed)21 日的利率決策/NP
12	央 行 / 總 裁 / 及 / 理 監 事 / 都 / 有 / 一 定 / 的 / 任 期 [Nc/Na/Caa/Na/D/V_2/A/DE/Na]	央行總裁及理監事/NP 都 有 一定 的任期/NP

6. Conclusion and Future Works

In this paper, we propose a Chinese shallow parser that can chunk Chinese sentences into five chunk types. We test the noise tolerance of the shallow parser and found that the accuracy of data with simulated unknown words only decreases slightly in chunk parsing. We also test our Chinese shallow parser on an open corpus, and found that it yields interesting chunking results.

Tolerance of unknown words is an essential characteristic of a Chinese shallow parser. In this paper, we demonstrate our parser's robustness in handling noisy data from proper nouns. However, we could not verify the robustness of chunking noisy data from other kinds of POS. Thus, adopting other POS systems, such as the Penn Chinese Treebank tagset, for Chinese shallow parsing could prove both

interesting and useful. In the future, we will improve our model by adding more types of noise, such as random noise, filled noise, and repeated noise proposed by Osborne [13]. In addition to Sinica Treebank, we will extend our training corpus by incorporating other corpora, such as Penn's Chinese Treebank.

Acknowledgements

This research was supported in part by the National Science Council under GRANT NSC94-2752-E-001-001-PAE.

References

1. Abney, S.P. Parsing by Chunks. in Berwick, R.C., Abney, S.P. and Tenny, C. eds. *Principle-Based Parsing: Computation and Psycholinguistics*, Kluwer, Dordrecht, 1991, 257-278.
2. Berger, A., Della Pietra, S.A. and Della Pietra, V.J. A maximum entropy approach to natural language processing. *Computational Linguistics*, *22*. 39-71.
3. Bikel, D.M., A Statistical Model for Parsing and Word-Sense Disambiguation. in *the Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora*, (Hong Kong, 2000), 155-168.
4. Chen, F.-Y., Tsai, P.-F., Chen, K.-J. and Huang, C.-R. 中文句結構樹資料庫的構建. *Computational Linguistics and Chinese Language Processing*, *4* (2). 87-104.
5. Chen, K.-J., Huang, C.-R., Chen, F.-Y., Luo, C.-C., Chang, M.-C., Chen, C.-J. and Gao, Z.-M. Sinica Treebank: Design Criteria, Representational Issues and Implementation. in Abeille, A. ed. *Treebanks Building and Using Parsed Corpora. Language and Speech series*, Kluwer, Dordrecht, 2003, 231-248.
6. Church, K.W., A Stochastic Parts Program and Noun Phrase Parser for Unrestricted Text. in *the Second Conference on Applied Natural Language Processing*, (1988), 136-143.
7. CKIP. Autotag, Academia Sinica, 1999.
8. Darroch, J.N. and Ratcliff, D. Generalized iterative scaling for log-linear models. *Annals of Mathematicl Statistics*, *43*. 1470-1480.
9. Li, B., Lu, Q. and Li, Y., Building a Chinese Shallow Parsed TreeBank for Collocation Extraction. in *CICLing*, (2003), 402-405.
10. Lu, Q., Zhou, J. and Xu, R.-F., Machine Learning Approaches for Chinese Shallow Parsers. in *International Conference On Machine Learning And Cybernetics*, (Xi'an, 2003), 2309- 2314.
11. Ma, W.-Y. and Chen, K.-J., A Bottom-up Merging Algorithm for Chinese Unknown Word Extraction. in *the Second SIGHAN Workshop on Chinese Language Processing*, (2003), 31-38.
12. Müller, F.H. and Ule, T., Annotating topological fields and chunks - and revising POS tags at the same time. in *Nineteenth International Conference on Computational Linguistics (COLING 2002)*, (Taipei, Taiwan, 2002), ACM, 695-701.

13. Osborne, M. Shallow Parsing using Noisy and Non-Stationary Training Material. *Journal of Machine Learning Research*, *2*. 695-719.
14. Ramshaw, L.A. and Marcus, M.P., Text chunking using transformation-based learning. in *The ACL Third Workshop on Very Large Corpora*, (1995), 82-94.
15. Tan, Y., Yao, T., Chen, Q. and Zhu, J., Applying Conditional Random Fields to Chinese Shallow Parsing. in *CICLing*, (2005), 167-176.
16. Tan, Y., Yao, T., Chen, Q. and Zhu, J., Chinese Chunk Identification Using SVMs plus Sigmoid. in *The First International Joint Conference on Natural Language Processing (IJCNLP-04)*, (2004), 527-536.
17. Viterbi, A.J. Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm. *IEEE Transactions on Information Theory*, *IT*. 260-269.
18. XIA, X. and WU, D., Parsing Chinese with an almost-context-free grammar. in *EMNLP-96, Conference on Empirical Methods in Natural Language Processing*, (Philadelphia, 1996).
19. Xu, R.-F., Lu, Q., Li, Y. and Li, W., The Construction of A Chinese Shallow Treebank. in *the Third SIGHAN Workshop on Chinese Language Processing*, (2004), 94-101.
20. Zhang, L.roach to Extract Chinese Chunk Candidates from Large Corpora. in *20th International Conference on Computer Processing of Oriental Languages (ICCPOL03)*, (ShenYang, P.R.China, 2003).
21. Zhao, T.-J., Yang, M.-Y., Liu, F., Yao, J.-M. and Yu, H., Statistics Based Hybrid Approach to Chinese Base Phrase Identification. in *Second Chinese Language Processing Workshop*, (Hong Kong, China, 2001), 73-77.
22. Zhou, M., A block-based robust dependency parser for unrestricted Chinese text. in *The second Chinese Language Processing Workshop attached to ACL2000*, (Hong Kong, 2000).