Sentence Parsing with Double Sequential Labeling in Traditional Chinese Parsing Task

Shih-Hung Wu, Hsien-You Hsieh
Chaoyang University of Technology
Wufeng, Taichung, Taiwan, ROC.

`shwu@cyut.edu.tw,`
`s10127602@gm.cyut.edu.tw`

Liang-Pu Chen
Institute for Information Industry
Taipei, Taiwan, ROC.

`eit@iii.org.tw`

Abstract

In this paper, we propose a new sequential labeling scheme, double sequential labeling, that we apply it on Chinese parsing. The parser is built with conditional random field (CRF) sequential labeling models. One focuses on the beginning of a phrase and the phrase type, while the other focuses on the end of a phrase. Our system, CYUT, attended 2012 the second CIPS-SGHAN conference Bake-off Task4, traditional Chinese parsing task, and got promising result on the sentence parsing task.

1 Introduction

Parsing is to identify the syntactical role of each word in a sentence, which is the starting point of natural language understanding. Thus, parser is an important technology in many natural language processing (NLP) applications. Theoretically, given a correct grammar, a parser can parse any valid sentence. However, in real world each writer might have a different grammar in mind; it is hard to parse all the sentences in a corpus without a commonly accepted grammar. PARSEVAL measures help to evaluate the parsing results from different systems in English (Harrison et al., 1991).

Parsing Chinese is even harder since it lacks of morphological markers on different part-of-speech (POS) tags, not to mention the different standards of word segmentation and POS tags. In 2012 CIPS-SGHAN Joint Conference on Chinese Language Processing, a traditional Chinese parsing task was proposed. The task was similar to the previous simplified Chinese parsing task (Zhou and Zhu, 2010), but it was with different evaluation set and standard. In this task, systems should recognize the phrase labels (S, VP, NP, GP, PP, XP, and DM), corresponding to Clause, Verb Phrase, Noun Phrase, Geographic Phrase, Preposition Phrase, Conjunction Phrase, and Determiner Measure phrase, all of which were defined in the User Manual of Sinica Treebank v3.0[1]. The goal of the task is to evaluate the ability of automatic parsers on complete sentences in real texts. The task organizers provide segmented corpus and standard parse tree. Thus, the task attenders can bypass the problem of word segmentation and the POS tag set problem, and focus on identifying the phrase boundary and type. The test set is 1,000 segmented sentences. Each sentence has more than 7 words, for example:

他 刊登 一則 廣告 在 報紙 上.

(He published an advertisement on newspaper in)
The system should recognize the syntactic structure in the given sentences, such as:

S(agent:NP(Nh:他) | Head:VC:刊登 | theme: NP (DM:一則 | Na: 廣告) | location: PP (P:在 | GP(NP(Na:報紙) | Ng:上))).

In additional to the sentence parsing task, there is a semantic role labeling task, which aims to find semantic role of a syntactic constituent. The participants can use either the training data provided by the organizers, which is called closed track, or the additional data, which is called open track.

In the following sections we will report how we use sequential labeling models on sentence chunking in the sentence parsing task in the closed track.

2 Methodology

Sequential labeling is a machine learning method that can train a tagger to tag a sequence of data.

---

[1] http://turing.iis.sinica.edu.tw/treesearch, page 6

The method is widely used in various NLP applications such as word segmentation, POS tagging, named entity recognition, and parsing. Applying the method to different tasks requires different adjustment; first at all is to define the tag set. On POS tagging task, the tag set is defined naturally, since each word will have a tag on it from the POS tag set. On other tasks, the tag set is more complex, usually including the beginning, the end, and outside of a sub-sequence. With an appropriate tag set, the tagging sequence can indicate the boundary and the type of a constituent correctly.

Our parsing approach is based on chunking (Abney, 1991) as in the previous Chinese parsing works (Wu et al. 2005, Zhou et al. 2010). Finkel et al. (2008) suggested CRF to train the model for parsing English. Since chunking only provides one level of parsing, not full parsing, several different approaches were proposed to achieve full parsing. Tsuruoka et al. (2009) proposed a bottom-up approach that the smallest phrases were constructed first, and merge into large phrases. Zhou et al. (2010) proposed another approach that maximal noun phrases were recognized first, and then decomposed into basic noun phrases later. Since one large NP often contains small NPs in Chinese, this approach can simplify many Chinese sentences. In this paper, we also define a double sequential labeling scheme to deal with the problem in a simpler way.

2.1 Sequential labeling

Many NLP applications can be achieved by sequential labeling. Input X is a data sequence to be labeled, and output Y is a corresponding label sequence. While each label Y is taken from a specific tag set. The model can be defined as:

$$p(Y \mid X) = \frac{1}{Z(X)} \exp(\sum_k \lambda_k f_k) \qquad (1)$$

where $Z(X)$ is the normalization factor, f_k is a set of features, λ_k is the corresponding weight. Many machine learning methods have been used on training the sequential labeling model, such as Hidden Markov Model, Maxima Entropy (Berger, 1996), and CRF (Lafferty, 2001). These models can be trained by a corpus with correct labeling and used as a tagger to label new input. The performance is proportional to the size of training set and counter proportional to the size of tag set. Therefore, if large training set is not available, decreasing the tag set can be a way to promote

the performance. In this task, we define two small tag sets for the closed task.

2.2 Double sequential labeling scheme

Sequential tagging can be used for labeling a series of words as a chunk by tagging them as the Beginning, or Intermediate of the chunk. The tagging scheme is call the B-I-O scheme. For the parsing task, we have to define two tags for each type of phrase, such as B-NP and I-NP for the noun phrase. The B-I-O scheme works well on labeling non-overlapping chunks. However, it cannot specify overlapping chunks, such as nested named entities, or long NP including short NPs.

In order to specify the overlapping chunks, we define a double sequential tagging scheme, which consists of two taggers, one is tagging the input sequence with I-B tags, and the other is tagging the input sequence with I-E tags, where E means the ending of some chunk. The first tagger can give the type and beginning position of each phrase in the sentence, while the second tagger can indicate the ending point of each phrase. Thus, many overlapping phrase can be specified clearly with this technology.

3 The Parsing Technology

The architecture of our system is shown in Figure 1. The system consists of three tagging modules and one post-processing module.
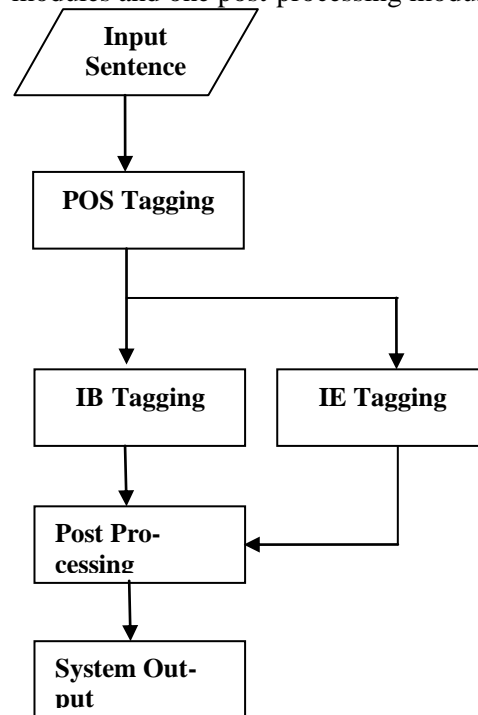

Figure 1. System architecture

The POS tagger will label each word in the input sentence with a POS tag. Then the sentence and the corresponding POS tags will be double labeled with beginning-or-intermediate-of-a-type and ending-or-not tag by the IB and IE taggers. A post-processing module will give the final boundary and the phrase type tag of the sentence. Each component will be described in the following subsections.

3.1 Part-of-Speech tagging

The POS tagging in our system is done by sequential labeling technology with CRF as in Lafferty (2001). We use the CRF++ toolkit[2] as our POS tagging tool. The model is trained from the official training set. We use the reduced POS tag set in our system. The tag set is the reduced POS tag set provided by CKIP. The complete set of POS tags is defined in CKIP[3]. Figure 2 shows the architecture of CRF tagger. For different applications, system developers have to update the tag set, feature set, preprocessing module and run the training process of the CRF model. Once the model is trained, it can be used to process input sentences with the same format.

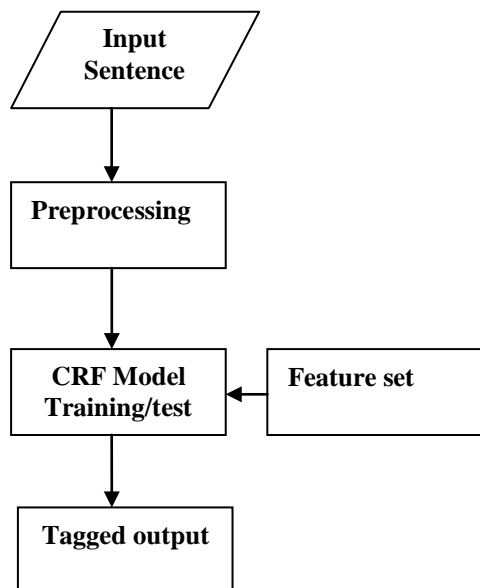The feature set for POS tagging is the word itself and the word preceding it and the word following it.

Figure 2. CRF tagger architecture

Preprocessing for POS tagging:

The training sentences have to be processed before they can be used as the input of CRF++ toolkit. Table 1 shows an example of the input format of training a CRF tagger. The original sentence in the training corpus is:
S(NP(Nh: 他 |DE: 的 |NP(NP(Na: 作 品)|Caa: 與 |NP(Na:生活|Na:情形)))|PP(P:被)|VG:拍成|Di: 了|NP(Na:電影)).

The first column shows the words in the sentence, the second column, which is for additional features, is not used in this case, and the third column is the POS tag. Since words in the DM phrases do not have POS tags in the training set, the tag DM itself is regarded as the POS tag for them.

Word	N/A	POS
他	NA	Nh
的	NA	DE
作品	NA	Na
與	NA	Caa
生活	NA	Na
情形	NA	Na
被	NA	P
拍成	NA	VG
了	NA	Di
電影	NA	Na

Table 1. A POS tagging training example

Table 2 shows the features used to train the POS tagger. In our system, due to the time limitation, the features are only the word itself, the word preceding it, and the word following it. Zhou et al. (2010) suggested that more features, such as more context words, prefix or suffix of the context words, might improve the accuracy of POS tagging.

Word Unigrams	W_{-1}, W_0, W_1

Table 2. Features used to train the POS tagger

3.2 Boundaries and types of constituents tagging

The POS tagging is not evaluated in this task, which is regarded as the feature preparation for parsing. The parsing result is based on both words and POS.

In our double sequential labeling scheme, every sentence will be labeled with two tags from

two tag set. The first tag set is the IB set, which consists of B, the beginning word, and I, the intermediate word, of all the types of phrases in the task, ie., S, NP, VP, and PP. Note that DM and GP were processed separately. The second tag set is the IE set, which consists of only E, the ending word of any phrase or I, other words.

The training sentences also have to be processed before they can be used as the input of CRF++ toolkit. Table 3 shows an example of the input format of training the BIO CRF tagger. The first column shows the words in the sentence, the second column is the corresponding POS, and the third column is the IB tag.

Word	POS	IB tag
他	Nh	B-NP
的	DE	I-NP
作品	Na	B-NP
與	Caa	I-NP
生活	Na	B-NP
情形	Na	I-NP
被	P	B-PP
拍成	VG	I-S
了	Di	I-S
電影	Na	B-NP

Table 3. An IB tagging training example

Table 4 shows an example of the input format of training the EO CRF tagger. The first column shows the words in the sentence, the second column is the corresponding POS, and the third column is the IE tag.

Word	POS	IE tag
他	Nh	I
的	DE	I
作品	Na	E
與	Caa	I
生活	Na	I
情形	Na	E
被	P	E
拍成	VG	I
了	Di	I
電影	Na	E

Table 4. An IE tagging training example

Table 5 shows the features used to train the double sequential labeling tagger. In our system, also due to the time limitation, the features are

the unigrams and bigrams of the word itself, the word preceding it, the word following it and the unigram, bigram, trigrams of the corresponding POSs of the context words. Zhou et al. (2010) suggested that the accuracy of tagging might be improved by more features, such as more context words, combination of POSs and words in the context.

Word Unigrams	W_{-1}、W_0、W_1
Word Bigrams	$W_{-1}W_0$、W_0W_1
POS Unigrams	P_{-1}、P_0、P_1
POS bigrams	$P_{-1}P_0$、P_0P_1
POS trigrams	$P_{-1}P_0P_1$

Table 5. Features used to train the double sequential labeling taggers

3.3 Post-processing to determine the boundaries and the types of constituents

After each word in the sentence is tagged with two tags, one from IB and one from IE, our system will determine the type and boundary of each phrase in the sentence. By integrating the information from both IB and IE labels, the boundary and type of phrases will be determined in the module.

Step 1: Combine the two labels to determine boundary. The B tags indicate the begging of a certain phrase. While the following I tags with the same phrase type indicate the intermediate of the same phrase. An I tag with different type or an E tag also indicates the end of a phrase. The type of the I tag which is different to the B tag will be stored for the next step.

Step 2: Put back the phrases with missing B tags during the step 1. The phrases contains I tag with different type will be labeled as a larger phrase with the type of the I tag.

Step 3: Add the GP phrase label according to the presentence of the Ng POS tag. Table 6 shows examples on how the post-processing works on GP. Phrases without ending tags will be tagged as ended at the last word.

Table 7 (at the end of the paper) shows a complete example.

S(agent:NP(Nh:我)|time:D:原本|Head:VF:打算|goal:VP(PP(P:在|GP(NP(Na:自然|Na:科學類)|Ng:中))|VC:找|NP(Na:答案)))
PP(Head:P: 當　|DUMMY:GP(VP(VC: 教

|goal:NP(Nh:她)|NP(Na:水|Na:字))|Ng:時))	
VP(concession:Cbb: 雖 |Head:VD: 帶 給 |theme:NP(Na: 人 們)|goal:NP(GP(NP(Na: 生 活)|Ng: 上)|VP(Dfa: 很 |VH: 大)|DE: 的 |Nv: 方 便))	

Table 6. When there is a word labeled Ng, our system will treated that phrase as NG.

4 Experiment results

The training set size is 5.8 MB, about 65,000 parsed sentences. The test set size is 55.4 KB, which consists of 1,000 sentences. The closed test on our POS tagging system is 96.80%. Since the official test does not evaluate POS, we cannot report the POS accuracy in open test.

4.1 Official test result

The official-run result of our system in 2012 Sighan Traditional Chinese Sentence Parsing task is shown in Table 8, and the detail of each phrase type is shown in Table 9. The Precision, Recall, and F1 are all above the baseline. The official evaluation required that the boundary and phrase label of a syntactic constituent must be completely identical with the standard. The performance metrics are similar to the metrics of PARSEVAL as suggested in (Black et al., 1991): Precision, Recall, F1 measure are defined as follows:

Precision = # of correctly recognized constituents / # of all constituents in the automatic parse.

Recall = # of correctly recognized constituents / # of all constituents in the gold standard parse.

F1 = 2*P*R / (P + R).

	Micro-averaging		
	Precision	Recall	F1
CYUT-Run1	0.6695	0.5781	0.6204
Stanford Parser (Baseline)	0.6208	0.5481	0.5822

	Macro-Averaging		
	Precision	Recall	F1
CYUT-Run1	0.6944	0.5999	0.6437
Stanford Parser (Baseline)	0.5885	0.5634	0.5757

Table 8. Sentence parsing result of our system

(Type)	(#Truth)	(#Parser)	(%Ratio)
S	1233	938	76.07
VP	679	187	27.54
NP	2974	1737	58.41
GP	26	9	34.62
PP	96	24	25
XP	0	0	N/A

Table 9. Detailed result of our system

5 Error analysis on the official test result

In the official test, there were 87 sentences that our system gave correct full parsing. We find that most of the sentences contain large NP chunks. Since our system tend to chunk large NP, these sentences are best parsed by our system.

For example, sentence no.339:

{S(最好康贈品包括買筆電送液晶螢幕), NP(最好康贈品), VP(最好康), VP(買筆電送液晶螢幕), NP(筆電), VP(送液晶螢幕), NP(液晶螢幕)}

and sentence no.580:

{S(台中日光溫泉會館執行董事張榮福表示), NP(台中日光溫泉會館執行董事張榮福), NP(台中日光溫泉會館執行董事), NP(台中日光溫泉會館)}

In the formal run, there were 14 sentences that our system labeled wrong. We will analyze the causes and find a way to improve, especially on the missing S, GP error, and PP error sentences.

5.1 Error analysis on the missing S tag sentences

Our system will give an S tag if there is at least on word tagged B-S or I-S. Therefore, if there is no word tagged with S, our system will miss the S tag.

Consider sentence no. 97, the parsing result of our system is:

VP(VC:摩根富林明|NP(Nc:台灣|Na:增長|Na:基金|Na:經理人|Na:葉鴻儒)|VC:分析)

System result:

{VP(摩根富林明台灣增長基金經理人葉鴻儒分析), NP(台灣增長基金經理人葉鴻儒)}

Ground Truth:

{S(摩根富林明台灣增長基金經理人葉鴻儒分析), NP(摩根富林明台灣增長基金經理人葉鴻儒), NP(摩根富林明台灣增長基金經理人), NP(摩根富林明台灣增長基金)}

The precision, recall, and F1 are all 0. The main reason that our system failed to chunk the right NP is our system cannot tag the POS of the named entity 摩根富林明 as Nb. Also, since the NP is not complete and the last word of the sentence is a verb, our system failed to label the S. Named entity recognition is a crucial component of word segmentation, POS tagging, and parsing.

5.2   Error analysis on GP

Consider sentence no. 13, the parsing result of our system is:

S(GP(D: 然 後 |NP(Nh: 我)|Ng: 後)|VC: 排 |NP(DM: 一 個 |Na: 青 年 |Na: 男 子 |Na: 飛 躍)|VP(Cbb:而|VC:起))

System result:

{S(然後我後排一個青年男子飛躍而起), GP(然後我後), NP(我), NP(一個青年男子飛躍), VP(而起)}

Ground Truth:

{S(然後我後排一個青年男子飛躍而起), NP(我後排一個青年男子), NP(我後排), VP(而起)}

The precision, recall, and F1 are 0.4, 0.5, and 0.4444 respectively. Our system reported an extra GP(然後我後). In this case, the error is caused by a wrong POS tagging error. The POS of '後' is not Ng. This case is hard to solve, since the CKIP online POS tagger also tag it as Ng. Our system will tag the phrase GP once the POS Ng appeared.

Consider sentence no. 43, the parsing result of our system is:

S(NP(Na: 司 法 院 |DM: 多 年)|VP(GP(Ng:來)|VL:持續|VP(VC:選派|NP(Na:法官)|PP(P:到 |NP(Nc:國外))|VC:進修|VC:學習)))

System result:

{S(司法院多年來持續選派法官到國外進修學習), NP(司法院多年), VP(來持續選派法官到國外進修學習), GP(來), VP(選派法官到國外進修學習), NP(法官), PP(到國外), NP(國外)}

Ground Truth:

{S(司法院多年來持續選派法官到國外進修學習), NP(司法院), GP(多年來), VP(選派法官到國外進修學習), NP(法官), VP(到國外進修學習), NP(國外), VP(進修學習)}

The precision, recall, and F1 are 0.5, 0.5, and 0.5. Our system found a wrong boundary of the GP(多年來). This is cause by another wrong boundary of VP.

Consider sentence no. 69, the parsing result of our system is:

VP(NP(S(NP(Na:總裁 |Nb:莊秀石)|VE:預估 |VP(Dfa: 最 |VH: 快)|NP(Na: 一 ○ 二 年)|Ncd: 底)|VB:完工))

System result:

{VP(總裁莊秀石預估最快一○二年底完工), NP(總裁莊秀石預估最快一○二年底完工), S(總裁莊秀石預估最快一○二年底), NP(總裁 莊秀石), VP(最快), NP(一○二年)}

Ground Truth:

{S(總裁莊秀石預估最快一○二年底完工), NP(總裁莊秀石), VP(最快一○二年底完工), VP(最快), GP(一○二年底), NP(一○二年)}

The precision, recall, and F1 are 0.5, 0.5, and 0.5. Our system missed the GP(一○二年底). Because the POS of '底' is tagged wrongly as Ncd, should be Ng. This case is hard, the CKIP online system segmented and tagged it differently as 一○二(Neu) 年底(Nd).

	#	%
Wrong boundary	11	42%
Wrong POS Ng	7	27%
Missing POS Ng	6	23%
Correct GP	9	35%

Table 10. Result analysis on the 26 GP in official test

5.3   Error analysis on PP

Consider sentence no. 53, the parsing result of our system is:

VP(NP(PP(P:如|NP(Na:簡易|Na:餐飲)|Neqa: 部分|D:可|VC:分包|PP(P:給|NP(VH:專業|Na:餐 飲|Na:業者))|VC:經營)))

System result:

{PP(如簡易餐飲部分可分包給專業餐飲業 者經營), , PP(給專業餐飲業者) }

Ground Truth:

{PP(如簡易餐飲部分), PP(給專業餐飲業者)}

The precision, recall, and F1 are 0.5, 0.5, and 0.5. In this case, the error is caused by the missing ending tag of the first PP.

Consider sentence no. 237, the parsing result of our system is:

S(NP(NP(Na: 周 傑 倫)|VA: 前 進 |Nc: 好 萊 塢 |Na:首作|Na:青蜂俠)|D:仍|PP(P:在|NP(VC:拍攝 |Na:階段)))

System result:

237 { PP(在拍攝階段) }

Ground Truth:

{no PP}

The precision, recall, and F1 are 0.6, 0.6, and 0.6. In this case, the ground truth does not include the PP(在拍攝階段). Because in this case, the POS of '在' is not P, should be VCL. This case is hard to solve, since the CKIP online POS tagger also tag it as P.

Consider sentence no. 673, the parsing result of our system is:

S(S(Nd:目前|NP(DM:這波|Na:物價|Na:跌勢)|VH:主要)|V_11:是|NP(Cbb:因|Nc:全球|Na:金融|Na:危機)|VP(Cbb:而|VC:起))

System result:

　{no PP}

Ground Truth:

　{ PP(因全球金融危機) }

The precision, recall, and F1 are 0.4, 0.5, and 0.4444 respectively. In this case, our system missed the PP(因全球金融危機). Because the POS of '因' is tagged as Cbb instead of P. This case is also hard to solve, since the CKIP online POS tagger also tag it as Cbb.

	#	%
Wrong boundary	24	25%
Wrong IB type	27	28%
Missing POS P	48	50%
Correct PP	24	25%

Table 11. Result analysis on the 96 PP in official test

5.4 Error analysis on NP and VP

We find that there are five types of error in the NP or VP chunking of our system result.
1. Error on the right boundary
2. Error on the left boundary
3. Missing the NP or VP type
4. A large phrase covered two or more small phrases with exactly substring.
5. Exchange on type labeling: NP into VP or VP into NP

Causes of the errors:
1. Error on the right boundary is caused by the error on IE tagging, one end tag is missing or labeled at a wrong word.
2. Error on the left boundary is caused by the error on IB tagging, one begin tag is labeled at a wrong word or an additional tag is tagged.
3. Missing type is caused by missing a begin tag of NP or VP.
4. In many sentences, there are two small NPs form a large NP. In this case, our system can

only recognize the large NP only, thus the short NPs are missing.
5. The type of begin tag is wrong.

In the following examples, on the top is the output of our system, on the bottom is the ground truth.

NP error type examples:

Error type 1:

5 {S(富蘭克林華美投信日前舉辦迎接投資新時代), NP(富蘭克林華美), VP(日前舉辦迎接投資新時代), VP(迎接投資新時代), NP(投資新時代)}
{S(富蘭克林華美投信日前舉辦迎接投資新時代), NP(富蘭克林華美投信), VP(迎接投資新時代), NP(投資新時代)}　　0.6 0.75 0.6667

Error type 2:

38 {NP(基隆市警察局外事課今年破獲一起人口販運集團案), S(基隆市警察局外事課今年破獲一起人口販運集團案), NP(基隆市警察局外事課今年破獲), NP(基隆市警察局外事課), NP(販運集團)}
{S(基隆市警察局外事課今年破獲一起人口販運集團案), NP(基隆市警察局外事課), NP(一起人口販運集團案), NP(人口販運集團案), NP(人口販運集團), NP(人口販運), NP(人口)}　　0.4 0.2857 0.3333

Error type 3:

42 {S(詳情可上神乎科技官網瞭解), NP(詳情), NP(神乎科技官網)}
{S(詳情可上神乎科技官網瞭解), NP(詳情), NP(神乎科技官網), NP(神乎科技)}　　1 0.75 0.8571

Error type 4:

1 {S(黨主席蔡英文元旦當天將到台東縣迎曙光), NP(黨主席蔡英文元旦當天), NP(黨主席蔡英文), PP(到台東縣), NP(台東縣), NP(曙光)}
{S(黨主席蔡英文元旦當天將到台東縣迎曙光), NP(黨主席蔡英文), NP(元旦當天), PP(到台東縣), NP(台東縣), NP(曙光)}　　0.8333 0.8333 0.8333

Error type 5:

7 {S(不景氣時期舉債反易債留子孫), NP(不景氣時期舉債), NP(易債), NP(子孫)}
{S(不景氣時期舉債反易債留子孫), VP(不景氣時期舉債), NP(不景氣時期), S(債留子孫), NP(債), NP(子孫)}　　0.5 0.3333 0.4

VP error type examples:

Error type 1:

31	{S(各球團需補助才請洋將實在說不過去), NP(各球團), VP(才請洋將), NP(洋將)}
	{S(各球團需補助才請洋將實在說不過去), NP(各球團), NP(補助), VP(才請洋將實在說不過去), NP(洋將), VP(實在說不過去)} 0.75 0.5 0.6

Error type 2:

82	{S(消防人員才能讓災損減到最低), NP(消防人員), VP(才能讓災損減到最低), NP(災損減到)}
	{S(消防人員才能讓災損減到最低), NP(消防人員), NP(災損), VP(減到最低), VP(最低)}　　0.5 0.4 0.4444

Error type 3:

82	{S(消防人員才能讓災損減到最低), NP(消防人員), VP(才能讓災損減到最低), NP(災損減到)}
	{S(消防人員才能讓災損減到最低), NP(消防人員), NP(災損), VP(減到最低), VP(最低)}　　0.5 0.4 0.4444

Error type 5:

7	{S(不景氣時期舉債反易債留子孫), NP(不景氣時期舉債), NP(易債), NP(子孫)}
	{S(不景氣時期舉債反易債留子孫), VP(不景氣時期舉債), NP(不景氣時期), S(債留子孫), NP(債), NP(子孫)}　　0.5 0.3333 0.4

The error analysis on NP:
We manually analyze the error cases and show the percentage of each error type in the following tables. The percentage in table 12 is defined as:
# of error cases / total # of NP in gold standard

Error type	#	%
1	265	8.92%
2	415	13.96%
3	673	22.63%
4	31	1.05%
5	59	1.99%
Correct	1730	58.41%

Table 12. Error distribution on NP

The error analysis on VP:
We manually analyze the error cases and show the percentage of each error type in the following table. The percentage in table 13 is defined as:
# of error cases / total # of VP in gold standard

Error type	#	%
1	31	4.57%
2	154	22.69%
3	362	53.32%
4	0	0%
5	59	8.06%
Correct	187	27.54%

Table 13. Error distribution on VP

By observing the two tables, we find that missing the begin tag is the major cause of error. To overcome the shortage, IB tagging accuracy is the most important issue. Since the wrong type labeling error is not very heavy, our system should label more begin tag in the future.

6　Conclusion and Future work

This paper reports our approach to the traditional Chinese sentence parsing task in the 2012 CIPS-SIGHAN evaluation. We proposed a new labeling method, the double labeling scheme, on how to use linear chain CRF model on full parsing task. The experiment result shows that our approach is much better than the baseline result and has average performance on each phrase type.

According to the error analysis above, we can find that many error cases of our system were caused by wrong POS tags and wrong boundary of PP phrase. POS tagging accuracy can be improved by adding more effective features, as in the previous works, and enlarging the training set. The boundary of PP phrase determination can also be improved by a larger training set and rules. Our system works best on S, and worst on PP and VP. The main reason of missing VP and PP is the error of POS tagging. Therefore, a better POS tagger will improve the worst part significantly. Complicated NP is known to be the highest frequent phrase in Chinese and cannot be represented in linear chain CRF model. Our system still fails to recognize many NPs. The system performance on NP can be improved by defining better representation of tag set.

Due to the limitation of time and resource, our system is not tested under different experimental settings. In the future, we will test our system with more feature combination on both POS labeling and sentence parsing.

References

Steven Abney. 1991. *Parsing by chunks*, Principle-Based Parsing, Kluwer Academic Publishers.

Adam L. Berger, Stephen A. Della Pietra, and Vincent J. Della Pietra. 1996. *A Maximum Entropy approach to Natural Language Processing*. Computational Linguistics, Vol. 22, No. 1., pp. 39-71.

E. Black; S. Abney; D. Flickenger; C. Gdaniec; R. Grishman; P. Harrison; D. Hindle; R. Ingria; F. Jelinek; J. Klavans; M. Liberman; M. Marcus; S. Roukos; B. Santorini; T. Strzalkowski. 1991. *A Procedure for Quantitatively Comparing the Syntactic Coverage of English Grammars*. In Speech and Natural Language workshop, Pacific Grove, California, USA, Feburay 1991.

Jenny Rose Finkel, Alex Kleeman, Christopher D. Manning. 2008. *Efficient, Feature-based, Conditional Random Field Parsing*, in Proceedings of ACL-08: HLT, pages 959–967, Columbus, Ohio, USA, June 2008.

Philip Harrison, Steven Abney, Ezra Black, Dan Flickinger, Ralph Grishman Claudia Gdaniec, Donald Hindle, Robert Ingria, Mitch Marcus, Beatrice Santorini, and Tomek Strzalkowski.1991. *Evaluating Syntax Performance of Parser/Grammars of English*. In Jeannette G. Neal and Sharon M. Walter, editors, Natural Language Processing Systems Evaluation Workshop, Technical Report RL-TR-91-362, pages 71-77.

John Lafferty, Andrew McCallum, and Fernando Pereira. *Conditional random fields: Probabilistic models for segmenting and labeling sequence data*. in Proceedings of 18th International Conference on Machine Learning, 2001.

Yoshimasa Tsuruoka, Jun'ichi Tsujii, Sophia Anaiakou. 2009. *Fast Full Parsing by Linear-Chain Conditional Random Fields*. In Proceedings of EACL'09, pages 790-798.

Shih-Hung Wu, Cheng-Wei Shih, Chia-Wei Wu, Tzong-Han Tsai, and Wen-Lian Hsu. 2005. *Applying Maximum Entropy to Robust Chinese Shallow Parsing*. in Proceedings of ROCLING 2005, pp 257-271.

Qiang Zhou; Jingbo Zhu. 2010. *Chinese Syntactic Parsing Evaluation*. in Proceedings of CIPS-SIGHAN Joint Conference on Chinese Language Processing, August 28-29, Beijing, China.

Qiaoli Zhou; Wenjing Lang; Yingying Wang; Yan Wang; Dongfeng Cai. 2010. *The SAU Report for the 1st CIPS-SIGHAN-ParsEval-2010*. in Proceedings of CIPS-SIGHAN Joint Conference on Chinese Language Processing, August 28-29, Beijing, China.

Words	他	的	作品	與	生活	情形	被	拍成	了	電影
POS	Nh	DE	Na	Caa	Na	Na	P	VG	Di	Na
BI	B-NP	I-NP	B-NP	I-NP	B-NP	I-NP	B-PP	I-S	I-S	B-NP
IE	I	I	E	I	I	E	E	I	I	E
Step 1	NP(Nh:他|DE:的|NP(Na:作品)|Caa:與|NP(Na:生活|Na:情形)|PP(P:被)|VG:拍成|Di:了|NP(Na:電影)|@S									
Step 2	S(NP(Nh:他|DE:的|NP(Na:作品)|Caa:與|NP(Na:生活|Na:情形)|PP(P:被)|VG:拍成|Di:了|NP(Na:電影)|@									
Step 3	S(NP(Nh:他|DE:的|NP(Na:作品)|Caa:與|NP(Na:生活|Na:情形)|PP(P:被)|VG:拍成|Di:了|NP(Na:電影)))									

Table 7. A complete example of the Post-processing steps