Segmentation of Chinese Long Sentences Using Commas

Mei xun Jin[1], Mi-Young Kim[2], Dongil Kim[3] and Jong-Hyeok Lee[4]

Graduate School for Information Technology[1], Div. of Electrical and Computer Engineering[24], Pohang University of Science and Technology
Advanced Information Technology Research Center(Altrc)
{meixunj[1],colorful[2],jhlee[4]}@postech.ac.kr

Language Engineering Institute[3]
Div. of Computer, Electronics and Telecommunications
Yanbian University of Science and Technology
dongil@ybust.edu.cn

Abstract

The comma is the most common form of punctuation. As such, it may have the greatest effect on the syntactic analysis of a sentence. As an isolate language, Chinese sentences have fewer cues for parsing. The clues for segmentation of a long Chinese sentence are even fewer. However, the average frequency of comma usage in Chinese is higher than other languages. The comma plays an important role in long Chinese sentence segmentation. This paper proposes a method for classifying commas in Chinese sentences by their context, then segments a long sentence according to the classification results. Experimental results show that accuracy for the comma classification reaches 87.1 percent, and with our segmentation model, our parser's dependency parsing accuracy improves by 9.6 percent.

1 Introduction

Chinese is a language with less morphology and no case marker. In Chinese, a subordinate clause or coordinate clause is sometimes connected without any conjunctions in a sentence. Because of these characteristics, Chinese has a rather different set of salient ambiguities from the perspective of statistical parsing (Levy and Manning, 2003). In addition, the work for clause segmentation is also rather different compared with other languages.

However, in written Chinese, the comma is used more frequently (Lin, 2000). In English, the average use of comma per sentence is 0.869 (Jones, 1996a)[1] ~1.04(Hill, 1996), and in Chinese it is 1.79[2], which is one and a half to two more times as it is used in English. In Korean, the comma is used even less than it is in English (Lin, 2000).

Since Chinese has less morphology and no case marker, and the comma is frequently used, the comma becomes an important cue for long Chinese sentence parsing. Because more commas may appear in longer sentences, the necessity of analyzing the comma also increases.

Some handbooks about standard Chinese grammars list ten to twenty uses of the comma, according to the context. Among these uses, is occurrence at the end of a clause[3] in a sentence (Lin, 2000). About 30% of commas are used to separate the clause from its main sentence or neighbor clause(s). If the comma appears at the end of a clause, the position can naturally be set as the clause segmentation point.

This paper proposes Chinese long sentence segmentation by classifying the comma. In section 2, related work in clause segmentation and punctuation processing is presented. Comma classification criteria are then introduced, and the classification model follows. Afterwards, some experimental results show how the proposed comma classification and long sentence segmentation are effective in Chinese parsing. Finally, a conclusion will be given.

[1] The frequency of comma per sentence is calculated as = Total frequency of commas/(Total frequency of full stop + Total frequency of Question mark), based on the punctuation statistics of Jone's Phd. thesis P56,57.
[2] The calculation is based on People's Daily Corpus'98.
[3] Clause in this paper, is a predicate with its full complements, subject, object(s). According to the type of a predicate and the context, subject or object may or may not appear. Adjunct of the predicate may or may not be included in the clause.

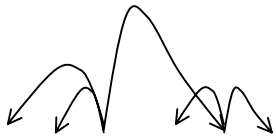

Figure1: example of a dependency relation

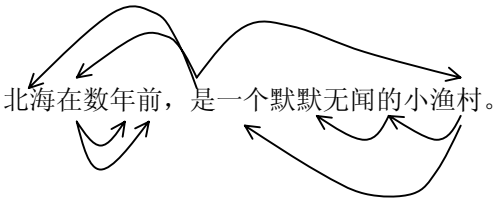

Figure2: example of a dependency relation

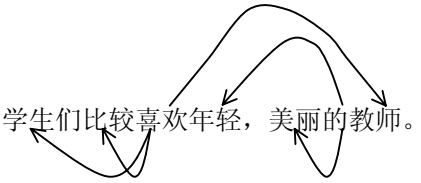

Figure 3: example of a dependency relation

Examples:

(1) 他们上午上课，下午做试验。
They have class in the morning and do experiments in the afternoon.

(2) 北海在数年前，是一个默默无闻的小渔村。
Several years ago, BeiHai City was only an unknown small fishing village.

(3) 学生们比较喜欢年轻，美丽的教师。
The students prefer young and beautiful teachers.

(4) 小明在写作业，妈妈在打毛衣。
Xiao Ming is doing homework and his mom is knitting.

(5) 尽管他很努力，但成绩不理想。
Though he studies very hard, his score is not satisfiable.

(6) 俄罗斯国内经济的发展变化，促进了两国之间的贸易往来。
The change of domestic economic development in Russia has promoted the trade exchange between two countries.

(7) 中国银行在去年十月，聘请日某公司做顾问。
Bank of China invited a Japanese company as its consoler last October.

(8) 在单位里，他是好领导，在家里，他是好爸爸。
He is a good leader in the company as well as a good daddy at home.

(9) 科研成果迅速转化为生产力，是这个开发区的特点。
The quick transfer of the scientific research achievement to industry is the characteristic of this development district.

(10) 学生们来到了操场，高高兴兴地。
The students happily come to the playground.

(11) 韩国对大连投资已连续三年增长，在大连，韩国投资企业受到各种优惠。
The investment from Korea to DaLian city has grown for three years, and all Korean investment companies in DaLian receive preferential treatment.

(12) 统计资料表明，大连对韩出口达一亿多美元。
The statistics show that the exportation from DaLian to Korea is reach to USD100,000,000.

(13) 一九九四年，通用在中国购买了四千多万美元的东西。
In 1994, TongYong Company purchased goods worthy of more than USD40,000,000.

(14) 她每天都起早，每天早上都要锻炼。
She gets up early, and does physical exercise every morning.

(15) 一号产品占据不到三成，二号产品比重达七成以上。
The occupation of the first products is less than 3/10 and the portion of the second ones is more than 7/10.

2 Related Work

2.1 Related Work for Clause Segmentation

Syntactic ambiguity problems increase drastically as the input sentence becomes longer. Long sentence segmentation is a way to avoid the problem. Many studies have been made on clause segmentation (Carreras and Marquez, 2002, Leffa, 1998, Sang and Dejean,2001). In addition, many studies also have been done on long sentences segmentation by certain patterns (Kim and Zhang, 2001, Li and Pei, 1990, Palmer and Hearst, 1997).

However, some researchers merely ignore punctuation, including the comma, and some researchers use a comma as one feature to detect the segmentation point, not fully using the information from the comma.

2.2 Related Work for Punctuation Processing

Several researchers have provided descriptive treatment of the role of punctuations: Jones (1996b) determined the syntactic function of the punctuation mark. Bayraktar and Akman (1998) classified commas by means of the syntax-patterns in which they occur. However, theoretical forays into the syntactic roles of punctuation were limited.

Many researchers have used the punctuation mark for syntactic analysis and insist that punctuation indicates useful information. Jones (1994) suc-

cessfully shows that grammar with punctuation outperforms one without punctuation. Briscoe and Carroll (1995) also show the importance of punctuation in reducing syntactic ambiguity. Collins (1999), in his statistical parser, treats a comma as an important feature. Shiuan and Ann (1996) separate complex sentences with respect to the link word, including the comma. As a result, their syntactic parser performs an error reduction of 21.2% in its accuracy.

Say (1997) provides a detailed introduction to using punctuation for a variety of other natural language processing tasks.

All of these approaches prove that punctuation analyses improve various natural language processing performance, especially in complex sentence segmentation.

3 Types of Commas

The comma is the most common punctuation, and the one that might be expected to have the greatest effect on syntactic parsing. Also, it seems natural to break a sentence at the comma position in Chinese sentences. The procedure for syntactic analysis of a sentence, including the segmentation part, is as follows:

1st step: segment the sentence at a comma
2nd step: do the dependency analysis for each segment
3rd step: set the dependency relation between segment pairs

In Chinese dependency parsing, not all commas are proper as segmentation points.

First, segmentation at comma in some sentences, will cause some of the words fail to find their heads. Figure 2 shows, in example (2), there are two words, 北海 (BeiHai City) and 在 (preposition) from the left segment have dependency relation with the word 是(is) of the right segment. So, the segmentation at comma , will cause two of words 北海 (BeiHai City) and 在 (preposition) in the left segment, cannot find their head in the second step of syntactic parsing stage.

Second, segmentation at commas can cause some words to find the wrong head. Example (3) of figure 3 shows two pairs of words with dependency relations. For each pair, one word is from the left segment, and one word is from the right segment : 喜欢 (like) from the left segment and 教师(teacher) from the right, 年轻 (young) from the left and 的 (of) from the right. Segmentation at the comma will cause the word 年轻(young) to get the word 喜欢 (like) as its head, which is wrong.

Example (2) and (3) demonstrate improper sentence segmentation at commas. In figure 2 and figure 3, there are two dependency lines that cross over the commas for both sentences. We call these kinds of commas *mul_dep_lines_cross comma* (multiple lines cross comma). In figure 1, there is only one dependency line cross over the comma. We call these kinds of commas *one_dep_line_cross comma*.

Segmentation at *one_dep_line_cross comma* is helpful for reducing parsing complexity and can contribute to accurate parsing results. However, we should avoid segmenting at the position of *mul_dep_lines_cross comma.* It is necessary to check each comma according to its context.

3.1 Delimiter Comma and Separator Comma

Nunberg (1990) classified commas in English into two categories, as a *delimiter comma* and a *separator comma*, by whether the comma is used to separate the elements of the same type[4] or not. While a *delimiter comma* is used to separate different syntactic types, a *separator comma* is used to separate members of conjoined elements. The commas in Chinese can also be classified into these two categories. The commas in example (3) and (4) are separators, while those in (2) and (5), are delimiters.

However, both delimiter comma and separator commas can be *mul_dep_line_cross commas.* In example (2), the comma is a delimiter comma as well as a *mul_dep_line_cross comma*. As a separator comma, the comma in example (3), is also a *mul_dep_line_cross comma*. Nunberg's classification cannot help to identify *mul_dep_line_cross commas.*

We therefore need a different kind of classification of comma. Both delimiter comma and separator comma can occur within a clause or at the end of a clause. Commas that appear at the end of a clause are clearly *one_dep_line_cross commas*. The segmentation at these kinds of comma is valid.

[4] Same type means that it has the same syntactic role in the sentence, it can be a coordinate phrase or coordinate clause.

3.2 Inter-clause Comma and Intra-clause Comma

Commas occurring within a clause are here called intra-clause commas. Similarly, commas at the end of a clause will be called inter-clause commas. Example (2), (3) include intra-clause commas, and example (4), (5) include inter-clause commas.

3.2.1 Constituents of the Two Segments Adjoining a Comma

A segment is a group of words between two commas or a group of words from the beginning (or end) of a sentence to its nearest comma.

To identify whether a comma is an inter-clause comma or an intra-clause comma, we assign values to each comma. These values reflect the nature of the two segments next to the comma. Either the left or right segment of a comma, can be deduced as a phrase[5], or several non-overlapped phrases, or a clause.(see examples (6)~(15)). The value we assign to a comma is a two-dimensional value (*left_seg, right_seg*). The value of *left_seg* and *right_seg* can be p(hrase) or c(lause), therefore the assigned value for each comma can be (p,p), (p,c), (c,p) or (c,c).

Commas with (p,p) as the assigned value, include the case when the left and right segment of the comma can be deduced as one phrase, as shown in example (6) or several non-overlapped phrases, as described in example (7).

We can assign the value of (c,p) to commas in example (8), (9) and (10), indicating the left adjoining segment is a clause and the right one is a phrase or several non-overlapped phrases. In a similar way, commas in example (11)~(13) are case of (p,c).

If a comma has (c,c) as the assigned value, both the left segment and the right segment can be deduced as a clause. The relation between the two clauses can be coordinate (example (14)) or subordinate (example (15)).

[5] Phrase is the group of words that can be deduced as the phrase in Chinese Penn Tree Bank 2.0. A phrase may contain an embedded clause as its adjunct or complement.
(a), 在他们写完作业之后，…
(b) 他们常去的饭店，…
In example (a) ,the PP has the embedded clause as its complement. And in example (b), the embedded clause is the adjunct of the NP.

3.2.2 Syntactic Relation between Two Adjoining Segments

A word (some words) in the left segment and a word (some words) in the right segment of a comma may or may not have a dependency relation(s). For a comma, if at least one word from the left segment has a dependency relation with a word from the right segment, we say the left segment and the right segment have a syntactic relation. Otherwise the two segments adjoining the comma have no syntactic relations. **Rel()** functions are defined in table-1.

Rel()
● To check if any words of the left segment has a dependency relation with the word of the right segment. ● If there is, Rel()=1 Otherwise Rel()=0.
Dir()
● To indicate how many direction(s) of the dependency relations the left and right segment have. when Rel()=1. ● For one_dep_line_cross comma, Dir()=1. ● For mul_dep_line_cross comma, if the directions of the dependency relations are the same, Dir()=1, else Dir()=2.
Head()
● To indicate which side of segment contains the head of any words of the other side, when Rel()=1. ● When Dir()=1, if the left segment contains any word as the head of a word of the right, Head() = left; Otherwise Head()=right. ● When Dir()=2, 1. According to the direction of dependency relation of these two segments, to find the word which has no head. 2. If the word is on the left, Head()=left, otherwise, Head()=right.

Table 1: functions Rel(), Dir() and Head()

For the *one_dep_line_cross comma*, the left and right segments have syntactic relation, and only one word from a segment has a dependency relation with a word from the other segment. For *mul_dep_line_cross comma,* at least two pairs of words from each segment have dependency relations. We then say that the left and right segments adjacent to the comma have multiple dependency relations. The directions of each relation may differ or not. We define a function **Dir()** as follows : if all the directions of the relations are the same, get 1 as

its value, else 2 for its value. This is in table-1. We also define function **Head()** to indicate whether the left segment or the right segment contains the head word of the other when the two segments have syntactic relation. This is also shown in table 1.

In example (3) as figure 3 shows, Rel()=1, Dir()=2 and Head()=left.

3.2.3 Inter-clause Comma and Intra-clause Comma

For commas assigned values (p,p) or (c,c), the function Rel() is always 1. Commas with values (c, p) or (p,c) can be further divided into two sub-cases. Table 2 shows the sub-case of (c,p), and table 3 shows the sub-cases of (p,c).

Rel() =0	The 2nd comma of Example (8);	(c,p)-I
Rel()=1	Example (9); Head() =right = p	
	Example (10); Head() = left=c	(c,p)-II

Table 2: sub-cases of commas with value of (c,p)

Rel()=0	Example (11);	(p,c)-I
Rel()=1	Example (12); Head() =left = p	
	Example (13); Head() = right=c	(p,c)-II

Table3: sub-cases of commas with value of (p,c)

Commas with the value of (p,p), (c,p)-II and (p,c)-II are used to connect coordinate phrases or to separate two constituents of a clause. These commas are intra-clause commas.

Commas with (c,c), (c,p)-I and (p,c)-I are used as a clause boundaries. These are inter-clause commas.

An inter-clause comma joins the clauses together to form a sentence. The commas that belong to an inter-clause category are safe as segmentation points (Kim, 2001).

4 Feature Selection

To identify the inter-clause or intra-clause role of a comma, we need to estimate the right and left segment conjuncts to the comma, using information from both segments. Any information to identify a segment as a clause or a phrase or phrases is useful. Carreras and Marquez (2001) prove that using features containing relevant information about a clause leads to more efficient clause identification. Their system outperforms all other systems in CoNLL'01 clause identification shared task (Sang & Dejean, 2001). Given this consideration, we select two categories of features as follows.

(1) Direct relevant feature category: predicate and its complements.

(2) Indirect relevant feature category: auxiliary words or adverbials or prepositions or clausal conjunctions.

Directly relevant features
VC: if a copula 是 appears
VA: if an adjective appears
VE: if 有 as the main verb appears
VV: if a verb appears
CS: if a subordinate conjunction appears
Indirectly relevant features
AD: if an adverb appears
AS: if an aspect marker appears
P: if a preposition appears
DE: if 的 appears
DEV:if 地 appears
DER: if 得 appears
BA_BEI: if 把 or 被 appears
LC: if a localizer appears
FIR_PR : if the first word is a pronoun
LAS_LO: if the last word is a localizer
LAS_T : if the last word is a time
LAS_DE_N : if the last word is a noun that follows 的
No_word : if the length of a word is more than 5
no_verb: if no verb(including VA)
DEC: if there is relative clause
ONE: if the segment has only one word

Table 4: feature types for classification

To detect whether a segment is a clause or phrase, the verbs are important. However, Chinese has no morphological paradigms and a verb takes various syntactic roles besides the predicate, without any change of its surface form. This means that information about the verb is not sufficient, in itself, to determine whether segment is a clause.

When the verb takes other syntactic roles besides the predicate, it's frequently accompanied by function words. For example, a verb can be used as the complement of the auxiliary word 地 or 的(Xia, 2000), to modify the following verb or noun. In these cases, the auxiliary words are helpful for deciding the syntactic role of the verb. Other function words around the verb also help us to estimate the

syntactic role of the verb. Under this consideration, we employ all the function words as features, where they are composed as the indirect relevant feature category.

Table 4 gives the entire feature set. The label of each feature type is same as the tag set of Chinese Penn Treebank 2.0 (see Xia (2000) for more detailed description). If the feature appears at the left segment, we label it as L_*feature type*, and if it is on the right, it's labeled as R_ *feature type*, where *feature type* is the feature that is shown on table 4.

The value for each feature is either 0 or 1. When extracting features of a sentence, if any feature in the table 4, appears in the sentence, we assign the value as 1 otherwise 0. The features of example (12) are extracted as table 5 describes. All of these values are composed as an input feature vector for comma classification.

L_VC =0	L_VA =0	L_VE =0
R_VC = 0	R_VA =0	R_VE =0
L_VV = 0	L_CS =0	L_AD =0
R_VV =1	R_CS =0	R_AD =0
L_AS =0	L_P =0	L_DE =0
R_AS =1	R_P =0	R_DE =1
L_DEV =0	L_DER = 0	L_BA_BEI =0
R_DEV =0	R_DER =0	R_BA_BEI=0
L_LC =0	L_DEC = 0	L_FIR_PR =0
R_LC =0	R_DEC =0	R_FIR_PR=0
L_LAS_LO =0	L_LAS_T =1	L_LAS_DE_N=0
R_LAS_LO=0	R_LAS_T=0	R_LAS_DE_N=1
L_No_word=0	L_no_verb =1	L_ONE = 0
R_No_word =1	R_no_verb =0	R_ONE =0

Table 5: the extracted features of example (12)

5 Experiments

For training and testing, we use the Chinese Penn Treebank 2.0 corpus based on 10-fold validation. First, using bracket information, we extract the type (inter-clause comma or intra-clause comma) for each comma, as we defined. The extracted information is used as the standard answer sheet for training and testing.

We extract the feature vector for each comma, and use support vector machines (SVM) to perform the classification work.

Performances are evaluated by the following four types of measures: accuracy, recall, $F_{\beta=1/2}$ for inter-clause and intra-clause comma respectively, and total accuracy. Each evaluation measure is calculated as follows.

$$\text{Inter(or intra)-clause comma accuracy}^{6} = \frac{\text{the number of correctly identified}}{\text{the number of identified}}$$

$$\text{Inter(or intra)-clause comma recall} = \frac{\text{the number of correctly identified}}{\text{total number of the class}}$$

$$\text{Inter(or intra)-clause comma } F_{\beta=1/2} = \frac{2 \times (\text{inter(or intra)} - \text{clause comma precision} \times \text{inter(or intra)} - \text{clause comma recall})}{(\text{inter(or intra)} - \text{clause comma precision} + \text{inter(or intra)} - \text{clause comma recall})}$$

$$\text{Total accuracy} = \frac{\text{total number of correctly identified}}{\text{total number of commas}}$$

5.1 Classification Using SVM

Support vector machines (SVM) are one of the binary classifiers based on maximum margin strategy introduced by Vapnik (Vapnik, 1995). For many classification works, SVM outputs a state of the art performance.

There are two advantages in using SVM for classification:

(1) High generalization performance in high dimensional feature spaces.
(2) Learning with combination of multiple features is possible via various kernel functions.

Because of these characteristics, many researchers use SVM for natural language processing and obtain satisfactory experimental results (Yamada, 2003).

In our experiments, we use SVM^{light} (Joachims, 1999) as a classification tool.

5.2 Experimental Results

First, we set the entire left segment and right segment as an input window. Table 6 gives the performance with different kernel functions. The RBF kernel function with γ =1.5 outputs the best performance. Therefore, in the following experiments, we use this kernel function only.

Next, we perform several experiments on how the selection of word window affects performance. First, we select the adjoining 3 words of the right and left segment each, indicated as win-3 in table 7.

[6] The inter-clause comma precision is abbreviated as inter-P. Same way, Inter-R for inter-clause comma recall, ..etc.

Second, we select the first 2 words and last 3 words of the left segment and the first 3 and last 2 of the right segment, indicated as win 2-3 in table 7. Finally, we use the part of speech sequence as input.

As the experimental results show, the part of speech sequence is not a good feature. The features with clausal relevant information obtain a better output. We also find that the word window of first 2-last 3 obtains the best total precision, better than using the entire left and right segments. From this, we conclude that the words at the beginning and end of the segment reveal segment clausal information more effectively than other words in the segment.

5.3 Comparison of Parsing Accuracy with and without Segmentation Model

The next experiment tests how the segmentation model contributes to parsing performance. We use a Chinese dependency parser, which was implemented with the architecture presented by Kim (2001) presents.

After integrating the segmentation model, the parsing procedure is as follows:

- Part of speech tagging.
- Long sentence segmentation by comma.
- Parsing based on segmentation.

Table 9 gives a comparison of the results of the original parser with the integrated parser.

5.4 Comparison with Related Work

Shiuan and Ann's (1996) system obtains the clues for segmenting a complex sentence in English by disambiguating the *link words*, including the comma. The approach to find the segmentation point by analyzing the specific role of the comma in the sentence seems similar with our approach. However, our system differs from theirs as follows:

(1) Shiuan and Ann's system sieves out just two roles for the comma, while ours gives an analysis for the complete usages of the comma.

(2) Shiuan and Ann's system also analyzes the clausal conjunction or subordinating preposition as the segmentation point.

Although the language for analysis is different, and the training and testing data also differ, the motivation of the two systems is the same. In addition, both systems are evaluated by integrating the original parser. The average accuracy of comma disambiguation in Shiuan and Ann's is 93.3% that is higher than ours by 6.2%. However, for parsing accuracy, Shiuan and Ann's system improves by 4%(error reduction of 21.2%), while ours improves by 9.6 percent.

Kernel function	Inter-P	Inter-R	Intra-P	Intra-R	Inter-F	Intra-F	Total-P
linear	74.22 %	77.87 %	72.52 %	70.61 %	76.00 %	71.56 %	73.14 %
Polynomial d=2	79.84 %	81.15 %	84.51 %	83.77 %	80.49 %	84.14 %	82.86 %
Polynomial d=3	78.57 %	81.15 %	88.39 %	86.84 %	79.84 %	87.61 %	84.86 %
RBF $\gamma = 0.5$	78.46 %	83.61 %	88.64 %	85.53 %	80.95 %	87.05 %	84.86 %
RBF $\gamma = 1.5$	78.69 %	78.69 %	**89.04 %**	**89.04 %**	78.69 %	**89.04 %**	**85.43 %**
RBF $\gamma = 2.5$	**80.62 %**	85.25 %	88.24 %	85.53 %	82.87 %	86.86 %	**85.43 %**
RBF $\gamma = 3.5$	79.41 %	**88.52 %**	85.05 %	79.82 %	**83.72 %**	82.35 %	82.86 %

Table 6: experimental results with different kernel functions

Word Window	Inter-P	Inter-R	Intra-P	Intra-R	Inter-F	Intra-F	Total-P
Win3	80.45 %	87.70 %	84.33 %	80.26 %	83.92 %	82.25 %	82.86 %
Win2-3	85.60 %	87.70 %	88.00 %	86.84 %	86.64 %	87.42 %	**87.14 %**

Table 7: experimental results for word window size

	Inter-P	Inter-R	Intra-P	Intra-R	Inter-F	Intra-F	Total-P
POS sequence	75.42 %	72.95 %	80.60 %	82.02 %	74.17 %	81.30 %	78.86 %

Table 8: experimental results for using part of speech sequence

	Original parser	Integrated parser
Average dependency parsing accuracy[7]	73.8%	83.4%
Average complete sentence accuracy	23.8%	25.4%

Table 9: comparison of parsing accuracy of the original parser with the integrated parser

[7] The evaluation measures are used as it is defined in Kim (2001).

6 Conclusion

In this paper, we propose a method to segment a Chinese sentence by classification of the comma.

We define the criteria for classification, and according to the criteria, a model for classification of the comma is given. The segmentation at the comma position seems to be efficient for improving the accuracy of dependency parsing by 9.6percent. Moreover, since commas more frequently appear in Chinese language, we expect our approach including salient and refined analysis of comma usages provides feasible solutions for segmentation.

However, the accuracy for the segmentation is not yet satisfactory. Since erroneous segmentation may cause a parsing failure for the entire sentence, errors can be serious. Further research should be done to improve the performance and reduce side effects for parsing the entire sentence.

Acknowledgments

This work was supported by the KOSEF through the Advanced Information Technology Research Center (AITrc), and by the BK21 Project.

References

M. Bayparktar, B. Say and V. Akman 1998, An analysis of English punctuation: the special case of comma, *International Journal of Corpus Linguistics*, 1998

X. Carreras, L. Marquez, V. Punyakanok, and D. Roth 2002, Learning and inference for clause identification, Proceeding of 13th European Conference on Machine Learning, Finland, 2002

R.L. Hill 1996, A comma in parsing: A study into the influence of punctuation (commas) on contextually isolated "garden-path" sentences. M.Phil disseration, Dundee University, 1996

T.Joachims 1999, Making large-Scale SVM Learning Practical. Advances in Kernel Methods - Support Vector Learning, B. Schölkopf and C. Burges and A. Smola (ed.), MIT-Press, 1999

B. Jones 1994, Exploring the role of punctuation in parsing natural text, Proceedings of COLING-94, pages 421-425

B. Jones 1996a, What's the point? A (computational) theory of punctuation, PhD Thesis, Centre for Cognitive Science, University of Edinburgh, Edinburgh, UK, 1996

B. Jones 1996b, Towards testing the syntax of punctuation, Proceeding of 34th ACL, 1996

M.Y.Kim, S.J. Kang, J.H. Lee 2001, Resolving ambiguity in Inter-chunk dependency parsing, Proceedings of the sixth Natural Language Processing Pacific Rim Symposium, Tokyo, Japan, 2001

S. Kim, B.Zhang and Y. Kim 2001, Learning-based intrasentence segmentation for efficient translation of long sentences, *Machine Translation*, Vol.16, no.3, 2001

Roger Levy and Christopher Manning. 2003. Is it harder to parse Chinese, or the Chinese Treebank? In Proceeding of ACL-2003.

V.J. Leffa 1998, clause processing in complex sentences, Proceeding of 1st International Conference on Language Resources and Evaluation, Spain,1998

W.C. Li, T.Pei, B.H. Lee and Chiou, C.F. 1990, Parsing long English sentences with pattern rules, Proceeding of 13th International Conference on Computational Linguistics, Finland, 1990

Shui-fang Lin 2000. *study and application of punctuation(标点符号的学习和应用)*. People's Publisher, P.R.China. (in Chinese)

Geoffrey Nunberg 1990. *the linguistics of punctuation* .CSLI lecture notes. 18, Stanford, California.

D.D. Palmer and M.A. Hearst 1997, Adaptive multilingual sentence boundary disambiguation, *Computational Linguistics*, Vol.27, 1997

E.F.T.K. Sang and H.Dejean. 2001, Introduction to the CoNLL-2001 shared task: clause identification, Proceeding of CoNLL-2001

B. Say and V. Akman 1997, current approaches to punctuation in computational linguistics, *Computers and the Humanities, 1997*

P.L. Shiuan and C.T.H. Ann 1996, A divide-and-conquer strategy for parsing, Proceedings of the ACL/SIGPARSE 5th international workshop on parsing technologies, Santa Cruz, USA, pp57-66

Fei Xia 2000, The bracketing Guidelines for the Penn Chinese Treebank(3.0)

Vladimir N Vapnik 1995 *The nature of statistical learning theory.* New York, 1995