Comparing Lexicalized Treebank Grammars Extracted from Chinese, Korean, and English Corpora

Fei Xia, Chung-hye Han, Martha Palmer, and Aravind Joshi
University of Pennsylvania
Philadelphia PA 19104, USA
{fxia,chunghye,mpalmer,joshi}@linc.cis.upenn.edu

Abstract

In this paper, we present a method for comparing Lexicalized Tree Adjoining Grammars extracted from annotated corpora for three languages: English, Chinese and Korean. This method makes it possible to do a quantitative comparison between the syntactic structures of each language, thereby providing a way of testing the Universal Grammar Hypothesis, the foundation of modern linguistic theories.

1 Introduction

The comparison of the grammars extracted from annotated corpora (i.e., Treebanks) is important on both theoretical and engineering grounds. Theoretically, it allows us to do a quantitative testing of the Universal Grammar Hypothesis. One of the major concerns in modern linguistics is to establish an explanatory basis for the similarities and variations among languages. The working assumption is that languages of the world share a set of universal linguistic principles and the apparent structural differences attested among languages can be explained as variation in the way the universal principles are instantiated. Comparison of the extracted syntactic trees allows us to quantitatively evaluate how similar the syntactic structures of different languages are. From an engineering perspective the extracted grammars and the links between the syntactic structures in the grammars are valuable resources for NLP applications, such as parsing, computational lexicon development, and machine translation (MT), to name a few.

In this paper we first briefly discuss some linguistic characteristics of English, Chinese, and Korean, and introduce the Treebanks for the three languages. We then describe a tool that extracts Lexicalized Tree Adjoining Grammars (LTAGs) from Treebanks and the results of its application to these three Treebanks. Next, we describe our methodology for automatic comparison of the extracted Treebank grammars, This consists primarily of matching syntactic structures (namely, templates and sub-templates) in each pair of Treebank grammars. The ability to perform this type of comparison for different languages has a definite positive impact on the possibility of sorting out the universal versus language-dependent features of languages. Therefore, our grammar extraction tool is not only an engineering tool of great value in improving the efficiency and accuracy of grammar development, but it is also very useful for investigating theoretical linguistics.

2 Three Languages and Three Treebanks

In this section, we briefly discuss some linguistic characteristics of English, Chinese, and Korean, and introduce the Treebanks for these languages.

2.1 Three Languages

These three languages belong to different language families: English is Germanic, Chinese is Sino-Tibetan, and Korean is Altaic (Comrie, 1987). There are several major differences between these languages. First, both English

and Chinese have predominantly subject-verb-object (SVO) word order, whereas Korean has underlying SOV order. Second, the word order in Korean is freer than in English and Chinese in the sense that argument NPs are freely permutable (subject to certain discourse constraints). Third, Korean and Chinese freely allow subject and object deletion, but English does not. Fourth, Korean has richer inflectional morphology than English, whereas Chinese has little, if any, inflectional morphology.

2.2 Three Treebanks

The Treebanks that we used in this paper are the English Penn Treebank II (Marcus et al., 1993), the Chinese Penn Treebank (Xia et al., 2000b), and the Korean Penn Treebank (Chung-hye Han, 2000). The main parameters of these Treebanks are summarized in Table 1.[1] The tags in each tagset can be classified into one of four types: (1) syntactic tags for phrase-level annotation, (2) Part-Of-Speech (POS) tags for head-level annotation, (3) function tags for grammatical function annotation, and (4) empty category tags for dropped arguments, traces, and so on.

We chose these Treebanks because they all use phrase structure annotation and their annotation schemata are similar, which facilitates the comparison between the extracted Treebank grammars. Figure 1 shows an annotated sentence from the Penn English Treebank.

3 LTAGs and Extraction Algorithm

In this section, we give a brief introduction to the LTAG formalism and to a system named LexTract, which we build to extract LTAGs from Treebanks.

[1]The reason why the average sentence length for Korean is much shorter than those for English and Chinese is that a big portion of the corpus for Korean Treebank includes dialogues that contain many one-word replies, whereas English and Chinese corpora consist of newspaper articles.

```
((S (PP-LOC (IN at)
            (NP (NNP FNX))
    (NP-SBJ-1 (NNS underwriters))
      (ADVP (RB still))
      (VP (VBP draft)
          (NP (NNS policies))
          (S-MNR
            (NP-SBJ (-NONE- *-1))
            (VP (VBG using)
              (NP
                (NP (NN fountain) (NNS pens))
                (CC and)
                (NP (VBG blotting) (NN papers))))))))
```

Figure 1: An example from Penn English Treebank

3.1 LTAG formalism

LTAGs are based on the Tree Adjoining Grammar formalism developed by Joshi, Levy, and Takahashi (Joshi et al., 1975; Joshi and Schabes, 1997). The primitive elements of an LTAG are elementary trees (*etrees*). Each *etree* is associated with a lexical item (called the *anchor* of the tree) on its frontier. LTAGs possess many desirable properties, such as the Extended Domain of Locality, which allows the encapsulation of all arguments of the anchor associated with an *etree*. There are two types of *etrees*: initial trees and auxiliary trees. An auxiliary tree represents a recursive structure and has a unique leaf node, called the *foot* node, which has the same syntactic category as the root node. Leaf nodes other than anchor nodes and foot nodes are *substitution* nodes. *Etrees* are combined by two operations: substitution and adjunction. The resulting structure of the combined *etrees* is called a *derived tree*. The combination process is expressed as a *derivation tree*. Figure 2 shows the *etrees*, the derived tree, and the derivation tree for the sentence *underwriters still draft policies*. Foot and substitution nodes are marked by $*$, and $\downarrow$, respectively. The dashed and solid lines in the derivation tree are for adjunction and substitution operations, respectively.

3.2 The Form of Target Grammars

Without further constraints, the *etrees* in the target grammar (i.e., the grammar to be extracted by LexTract) could be of various shapes. LexTract recognizes three types of

Language	corpus size (words)	average sentence length	# of POS tags	# of syntactic tags	# of function tags	# of empty category tags
English	1,174K	23.85 words	36	26	20	12
Chinese	100K	23.81 words	34	25	26	7
Korean	30K	10.52 words	17	18	17	4

Table 1: Size of the Treebanks and the tagsets used in each Treebank

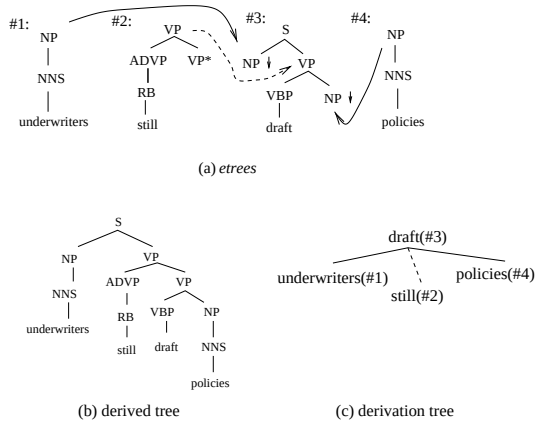


Figure 2: *Etrees*, derived tree, and derivation tree for *underwriters still draft policies*

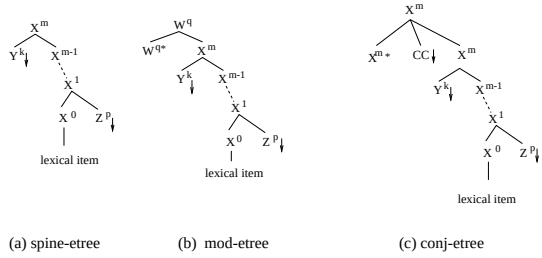


Figure 3: Three types of elementary trees in the target grammar

relation (namely, predicate-argument, modification, and coordination relations) between the anchor of an *etree* and other nodes in the *etree*, and imposes the constraint that all the *etrees* to be extracted should fall into exactly one of the three patterns in Figure 3.

- The spine-etrees for predicate-argument relations. X^0 is the head of X^m and the anchor of the *etree*. The *etree* is formed by a spine $X^m \rightarrow X^{m-1} \rightarrow .. \rightarrow X^0$ and the arguments of X^0.

- The mod-etrees for modification relations. The root of the *etree* has two children, one is a foot node with the label W^q, and the other node X^m is a modifier of the foot node. X^m is further expanded into a spine-etree whose head X^0 is the anchor of the whole mod-etree.

- The conj-etrees for coordination relations. In a conj-etree, the children of the root are two conjoined constituents and a node for a coordination conjunction. One conjoined constituent is marked as the foot node, and the other is expanded into a spine-etree whose head is the anchor of the whole tree.

Spine-etrees are initial trees, whereas mod-etrees and conj-etrees are auxiliary trees.

3.3 Extraction algorithm

The core of LexTract is an extraction algorithm that takes a Treebank sentence such as the one in Figure 1 and Treebank-specific information provided by the user of LexTract, and produces a set of *etrees* as in Figure 4 and a derivation tree. We have described LexTract's architecture, its extraction algorithm, and its applications in (Xia, 1999; Xia et al., 2000a). Therefore, we shall not repeat them in this paper other than pointing out that LexTract is completely language-independent.

3.4 Experiments

The results of running LexTract on English, Chinese, and Korean Treebanks are shown in Table 2. *Templates* are *etrees* with the lexical items removed. For instance, #3, #6, and #9 in Figure 4 are three distinct *etrees* but they share the same *template*.

Figure 5 shows the log frequency of templates in the English Treebank and percentage of template tokens covered by template

	template types	*etree* types	word types	*etree* types per word type	*etree* types per word token	CFG rules (unlexicalized)
Eng G_1	6926	131,397	49,206	2.67	34.68	1524
Ch G_2	1140	21,125	10,772	1.96	9.13	515
Kor G_3	634	9,787	6,747	1.45	2.76	177

Table 2: Grammars extracted from three Treebanks

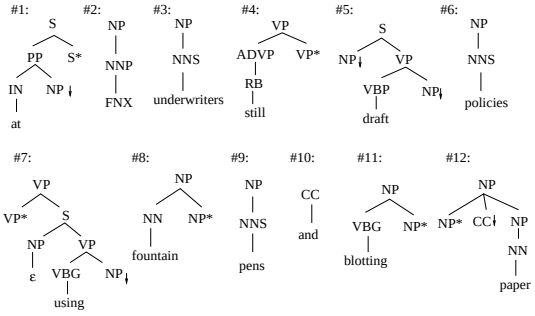
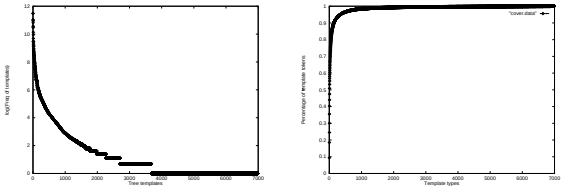


Figure 4: The extracted *etrees* from the fully bracketed *ttree*



(a) Frequency (b) Coverage

Figure 5: *Etree* template types and template tokens in the Penn English Treebank
(X-axes: (a) and (b) template types
Y-axes: (a) log frequency of templates; (b) percentage of template token covered by template types)

types.[2] In both cases, template types are sorted according to their frequencies and plotted on the X-axis. The figure shows that a small subset of template types, which occurs very frequently in the Treebank and can be seen as the core of the Treebank grammar, covers the majority of template tokens in the Treebank. For instance, the most frequent template type covers 9.37% of the template tokens and the top 100 (500, 1000 and 1500, respectively) template types cover 87.1% (96.6%, 98.4% and 99.0%, respectively) of the tokens, whereas about half (3440) of the template types occur once, accounting for only 0.32% of template tokens in total.

4 Comparing Three Treebank Grammars

In this section, we describe our methodology for comparing Treebank grammars and the experimental results.

4.1 Methodology

To compare Treebank grammars, we need to ensure that the Treebank grammars are based on the same tagset. To achieve that, we first create a new tagset that includes all the tags from the three Treebanks. Then we merge some tags in this new tagset into a single tag. This step is necessary because certain distinctions among some tags in one language do not exist in another language. For example, the English Treebank has distinct tags for verbs in past tense, past participals, gerunds, and so on; however, no such distinction is morphologically marked in Chinese and, therefore, the Chinese Treebank uses the same tag for verbs regardless of the tense and aspect. To make the conversion straightforward for verbs, we use a single tag for verbs in the new tagset. Next, we replace the tags in the original Treebanks with the tags in the new tagset, and then re-run LexTract to build Treebank grammars from those Treebanks.

Now that the Treebank grammars are based on the same tagset, we can compare them according to the templates and sub-templates that appear in more than one Treebank — that is, given a pair of Treebank grammars, we first calculate how many templates occur in both grammars;[3] Next, we decompose

[2] If a template occurs n times in the corpus, it is counted as one template type but n template tokens.

[3] Ideally, to get more accurate comparison results, we would like to compare *etrees*, rather than templates (which are non-lexicalized); however, comparing *etrees* requires bilingual parallel corpora, which we are cur-

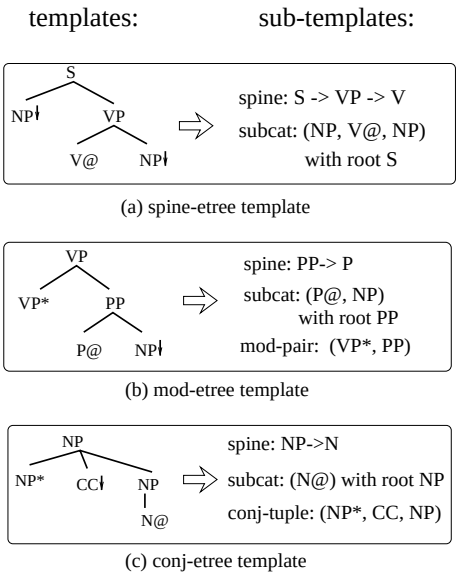


Figure 6: The decompositions of *etree* templates (In sub-templates, @ marks the anchor in subcategorization frame, * marks the modifiee in a modifier-modifiee pair.)

each template into a list of *sub-templates* (e.g., spines and subcategorization frames) and calculate how many of those sub-templates occur in both grammars. A template is decomposed as follows: A spine-etree template is decomposed into a spine and a subcategorization frame; a mod-etree template is decomposed into a spine, a subcategorization frame, and a modifier-modifiee pair; a conj-etree template is decomposed into a spine, a subcategorization frame, and a coordination tuple. Figure 6 shows examples of this decomposition for each type of template.

4.2 Experiments

After tags in original Treebanks being replaced with the tags in the new tagset, the numbers of templates in the new Treebank grammars decrease by about 50%, as shown in the second column of Table 3 (cf. the second column in Table 2). Table 3 also lists the numbers of sub-templates, such as spines and subcategorization frames, for each grammar.

Table 4 lists the numbers of template types shared by each pair of Treebank grammars and the percentage of the template tokens in each Treebank which are covered by these common template types. For example, there are 237 template types that appear in both English and Chinese Treebank grammars. These 237 template types account for 80.1% of template tokens in the English Treebank, and 81.5% of template tokens in the Chinese Treebank. The table shows that, although the number of matched templates are not very high, they are among the most frequent templates and they account for the majority of template tokens in the Treebanks. For instance, in the (Eng, Ch) pair, the 237 template types that appear in both grammars is only 7.5% of all the English template types, but they cover 80.1% of template tokens in the English Treebank. If we define the core grammar of a language as the set of the templates that occur very often in the Treebank, the data suggest that the majority of the core grammars are easily inter-mappable structures for these three languages.

If we compare sub-templates, rather than templates, in the Treebank grammars, the percentages of matched sub-template tokens (as in Table 5) are higher than the percentages of matched template tokens. This is because two distinct templates may share common sub-templates.

4.3 Unmatched templates

Our previous experiments (see Table 4) show that the percentages of unmatched template tokens in three Treebanks range from 16.0% to 43.8%, depending on the language pairs. Given a language pair, there are many possible reasons why a template appears in one Treebank grammar, but not in the other. We divide those unmatched templates into two categories: spuriously unmatched templates and truly unmatched templates.

Spuriously unmatched templates *Spuriously* unmatched templates are templates that either should have found a matched template in the other grammar or should not have been created by LexTract in the first place if the Treebanks were complete, uniformly annotated, and error-free. A spuriously unmatched template exists because of one of the

rently building.

	templates	*subtemplates*				
		spines	subcat frames	mod-pairs	conj-tuples	total
Eng	3139	500	541	332	53	1426
Ch	547	108	180	152	18	458
Kor	271	55	58	53	6	172

Table 3: Treebank grammars with the new tagset

		matched templates	templates with unique tags	other unmatched templates
(Eng,Ch)	type (#)	(237, 237)	(536, 99)	(2366, 211)
	token (%)	(80.1, 81.5)	(2.8, 12.3)	(17.1, 6.2)
(Eng, Kor)	type (#)	(83, 83)	(2075, 6)	(981, 182)
	token (%)	(57.7, 82.8)	(28.1, 0.1)	(14.2, 17.1)
(Ch, Kor)	type (#)	(59,59)	(324,6)	(164, 206)
	token (%)	(57.2, 84.0)	(29.4, 0.1)	(13.4, 16.0)

Table 4: Comparisons of templates in three Treebank grammars

following reasons:

(S1) Treebank size: The template is linguistically sound in both languages, and, therefore, should belong to the grammars for these languages. However, the template appears in only one Treebank grammar because the other Treebank is too small to include such a template. Figure 7(S1) shows a template that is valid for both English and Chinese, but it appears only in the English Treebank, not in the Chinese Treebank.

(S2) Annotation difference: Treebanks may choose different annotations for the same constructions; consequentially, the templates for those constructions look different. Figure 7(S2) shows the templates used in English and Chinese for a VP such as "*surged 7 (dollars)*". In the template for English, the *QP* projects to an NP, but in the template for Chinese, it does not.

(S3) Treebank annotation error: A template in a Treebank may result from annotation errors in that Treebank. If no corresponding mistakes are made in the other Treebank, the template in the first Treebank will not match any template in the second Treebank. For instance, in the English Treebank the word *about* in the sentence *About 5 people showed up* is often mis-tagged as a preposition, resulting in the template in Figure 7(S3). Not surprisingly, that template does not match any template in the Chinese Treebank.

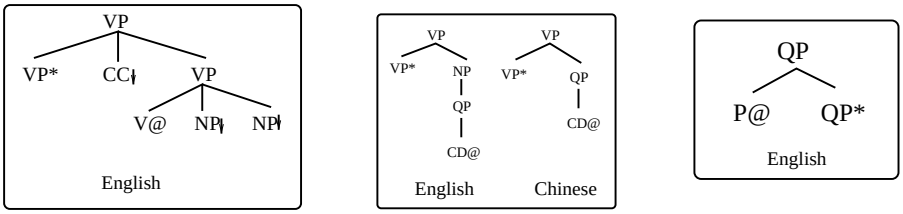
Truly unmatched templates A *truly* unmatched template is a template that does not match any template in the other Treebank even if we assume both Treebanks are perfectly annotated. Here, we list three reasons why a truly unmatched template exist.

(T1) Word order: The word order determines the positions of arguments w.r.t. their heads, and the positions of modifiers w.r.t. their modifiees. If two languages have different word orders, their templates which include arguments of a head or a modifier are likely to look different. For example, Figure 8(T1) show the templates for transitive verbs in Chinese and Korean grammars. The templates do not match because of the different positions of the object of the verb.

(T2) Unique tags: For each pair of languages, some Part-of-speech tags and syntactic tags may appear in only one language. Therefore, the templates with those tags will not match any templates in the other language. For instance, in Korean the counterparts of preposition phrases in English and Chinese are noun phrases (with postpositions attaching to them, not preposition phrases); therefore, the templates with PP in Chinese,

		spines	subcat frames	mod-pairs	conj-tuples	total
(Eng,Ch)	type	(60,60)	(92, 92)	(83,83)	(11,11)	(246,246)
	token	(94.7,87.2)	(94.0, 86.3)	(82.6, 80.0)	(84.2, 99.1)	(91.4, 85.2)
(Eng, Kor)	type	(39, 39)	(40, 40)	(46, 46)	(1, 1)	(126,126)
	token	(70.3, 96.9)	(62.1, 96.6)	(56.8, 99.5)	(9.3, 52.3)	(63.4,97.3)
(Ch, Kor)	type	(28, 28)	(25,25)	(29,29)	(1, 1)	(83, 83)
	token	(74.2, 99.2)	(63.1, 98.1)	(60.2, 93.4)	(0.1, 0.4)	(66.1, 96.9)

Table 5: Comparisons of sub-templates in three Treebank grammars



(S1) Treebank size (S2) annotation difference (S3) annotation error

Figure 7: Examples of spuriously unmatched templates

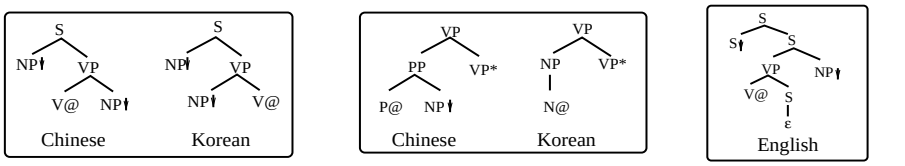
such as the left one in Figure 8(T2), do not match any template in Korean.

(T3) Unique syntactic relations: Some syntactic relations may be present in only one of the pair of languages being compared. For instance, the template in Figure 8(T3) is used for the sentence such as *"You should go," said John*, where the subject of the verb *said* appears after the verb. No such template exists in Chinese.

So far, we have listed six possible reasons for unmatched templates. Without manually examining all the unmatched templates, it is difficult to tell how many unmatched templates are caused by a particular reason. Nevertheless, these reasons help us to interpret the results in Table 4. For instance, the table shows that Korean grammars cover only 57.7% of template tokens in the English Treebank, and 57.2% in the Chinese Treebank, whereas the coverages for other language pairs are all above 80%. We suspect that this difference of coverage is mainly caused by (S1), (T1), and (T2). That is, first, Korean Treebank is much smaller than the English and the Chinese Treebanks, English and Chinese Treebanks may have many tree templates that simply was not found in the Korean Treebank; Second, English and Chinese are predominantly head-initial, whereas Korean is head-final, therefore, many templates in English and Chinese can not find matched templates in Korean because of the word order difference; Third, Korean does not have preposition phrases, causing all the templates in English and Chinese with PPs become unmatched. To measure the effect of the word order factor to the matching rate, we re-did the experiment in Section 4.2, but this time we ignored the word order — that is, we treat templates as unordered trees. The results are given in Table 6. Comparing this table with Table 4, we can clearly see that, the percentages of matched templates increase substantially for (Eng, Kor) and (Ch, Kor) when the word order is ignored. Notice that the matching percentage for (Eng, Ch) does not change as much because the word orders in English and Chinese are much similar than the orders in English and Korean.

5 Conclusion

We have presented a method of quantitatively comparing LTAGs extracted from Treebanks. Our experimental results show a high proportion of easily inter-mappable structures, giving a positive implications for Universal Grammar hypothesis. We have also described a number of reasons why a particular tem-



(T1) word order (T2) unique tags (T3) unique relation

Figure 8: Truly unmatched templates

		matched templates	tag mismatches	other mismatches
(Eng,Ch)	type	(334, 259)	(536, 99)	(2269, 189)
	token	(82.8, 82.2)	(2.8, 12.3)	(14.4, 5.5)
(Eng, Kor)	type	(222, 167)	(2075, 6)	(842, 98)
	token	(66.4, 92.4)	(28.1, 0.1)	(5.5, 7.5)
(Ch, Kor)	type	(126,125)	(324,6)	(97, 140)
	token	(68.3, 97.3)	(29.4, 0.1)	(2.3, 2.6)

Table 6: Comparisons of templates w/o orders

plate does not match any template in other languages and tested the effect of word order on matching percentages.

There are two natural extensions of this work. First, running an alignment algorithm on parallel bracketed corpora to produce word-to-word mappings. Given such word-to-word mappings and our template matching algorithm, we can automatically create lexicalized *etree*-to-*etree* mappings, which can be used for semi-automatic transfer lexicon construction. Second, LexTract can build derivation trees for each sentence in the corpora. By comparing derivation trees for parallel sentences in two languages, instances of structural divergences (Dorr, 1993; Dorr, 1994; Palmer et al., 1998) can be automatically detected.

References

Chung-hye Han. 2000. Bracketing Guidelines for the Penn Korean Treebank (draft). www.cis.upenn.edu/ xtag/korean.tag.

Bernard Comrie. 1987. *The World's Major Languages*. Oxford University Press, New York.

B. J. Dorr. 1993. *Machine Translation: a View from the Lexicon*. MIT Press, Boston, Mass.

B. J. Dorr. 1994. Machine translation divergences: a formal description and proposed solution. *Computational Linguistics*, 20(4):597–635.

Aravind Joshi and Yves Schabes. 1997. Tree Adjoining Grammars. In A. Salomma and G. Rosenberg, editors, *Handbook of Formal Languages and Automata*. Springer-Verlag, Herdelberg.

Aravind K. Joshi, L. Levy, and M. Takahashi. 1975. Tree Adjunct Grammars. *Journal of Computer and System Sciences*.

M. Marcus, B. Santorini, and M. A. Marcinkiewicz. 1993. Building a Large Annotated Corpus of English: the Penn Treebank. *Computational Lingustics*.

Martha Palmer, Owen Rambow, and Alexis Nasr. 1998. Rapid Prototyping of Domain-Specific Machine Translation System. In *Proc. of AMTA-1998*, Langhorne, PA.

Fei Xia, Martha Palmer, and Aravind Joshi. 2000a. A Uniform Method of Grammar Extraction and its Applications. In *Proc. of Joint SIGDAT Conference on Empirical Methods in Natural Language Processing and Very Large Corpora (EMNLP/VLC)*.

Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Okurowski, John Kovarik, Shizhe Huang, Tony Kroch, and Mitch Marcus. 2000b. Developing Guidelines and Ensuring Consistency for Chinese Text Annotation. In *Proc. of the 2nd International Conference on Language Resources and Evaluation (LREC-2000)*, Athens, Greece.

Fei Xia. 1999. Extracting Tree Adjoining Grammars from Bracketed Corpora. In *Proc. of 5th Natural Language Processing Pacific Rim Symposium (NLPRS-99)*, Beijing, China.