Automatically Extracting and Comparing Lexicalized Grammars for Different Languages

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 quantitative comparison between the syntactic structures of three languages: English, Chinese and Korean. This is made possible by first extracting Lexicalized Tree Adjoining Grammars from annotated corpora for each language and then performing the comparison on the extracted grammars. We found that the majority of the core grammar structures for these three languages are easily inter-mappable.

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 the establishment of 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 the Treebanks and the results of its application. Next, we describe our methodology for automatic comparison of the extracted Treebank grammars, which consists primarily of matching syntactic structures (namely, templates, sub-templates and context-free rules) in each pair of Treebank grammars. The ability to perform this type of comparison for different languages enables us to distinguish language-independent features from language-dependent ones. 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 Our Annotated Corpora

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

2.1 Differences between the 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 Treebank Description

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 [Han *et al.*, 2001]. The main parameters of these Treebanks are summarized in Table 1.[1] The tagsets include four types of tags: Part-Of-Speech (POS) tags for head-level annotation, syntactic tags for phrase-level annotation, function tags for grammatical function annotation, and 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 English Penn Treebank.

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

Language	corpus size (words)	ave sentence length (words)	tagset size
English	1,174K	23.85	94
Chinese	100K	23.81	92
Korean	54K	10.71	61

Table 1: Sizes of the Treebanks and their tagsets

```
((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 the English Penn Treebank

3 Extracting Grammars

In this section, we give a brief introduction to the LTAG formalism and to a system named LexTract, which was built to extract LTAGs from the Treebanks [Xia, 1999; Xia *et al.*, 2000a].

3.1 The Grammar Formalism

LTAGs are based on the Tree Adjoining Grammar formalism developed by Joshi and his colleagues [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 history of 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 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 relations between the anchor of an *etree* and other nodes in the *etree*; namely, predicate-argument, modification, and coordination relations. It imposes the constraint that all the *etrees* to be

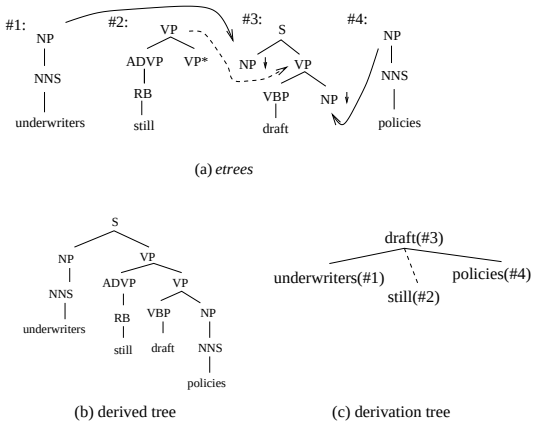


Figure 2: *Etrees*, derived tree, and derivation tree for *underwriters still draft policies*

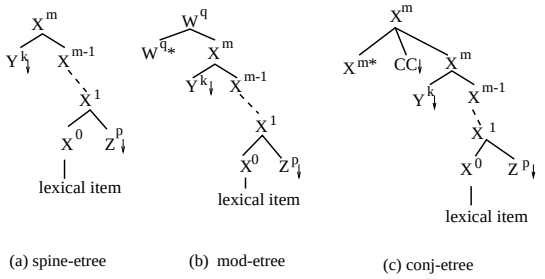


Figure 3: Three types of elementary trees in the target grammar

extracted should fall into exactly one of the three patterns (as in Figure 3):[2]

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 the spine $X^m \rightarrow X^{m-1} \rightarrow .. \rightarrow X^0$ and the arguments of X^0.

Mod-etrees for modification relations: The root of the *etree* has two children, one is a foot node with the same label W^q as the root node, 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.

Conj-etrees for coordination relations: In a conj-etree, the children of the root are two conjoined constituents and a node for a coordinating 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 by themselves are initial trees, whereas mod-etrees and conj-etrees are auxiliary trees.

3.3 The LexTract 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

[2]The precedence relation between the children of the nodes in these three patterns is unspecified, and may vary from language to language.

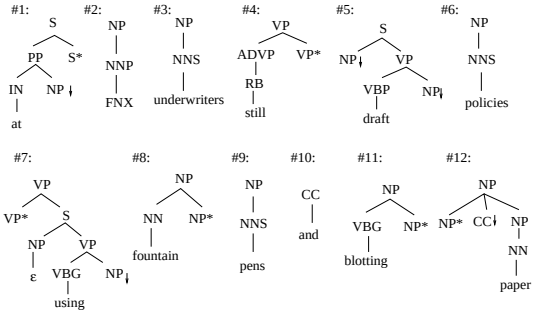


Figure 4: The extracted *etrees* from the phrase structure in Figure 1

produces a set of *etrees* as in Figure 4 and a derivation tree. LexTract's extraction algorithm has been described in [Xia, 1999] and is completely language-independent. It has been successfully applied to the development of language processing tools such as SuperTaggers [Xia *et al.*, 2000a] and statistical LTAG parsers [Sarkar, 2001].

3.4 Extracted Grammars

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*. LexTract is designed to extract LTAGs, but simply reading context-free rules off the templates in an extracted LTAG will yield a context-free grammar. The last column in the table shows the numbers of the non-lexicalized context-free rules.

In each Treebank, a small subset of template types, which occur very frequently in the Treebank and can be seen as members of the core of the Treebank grammar, covers the majority of template tokens in the Treebank. For instance, the top 100 (500, 1000 and 1500, respectively) template types in the English Penn Treebank 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 the template tokens in total.

	template types	*etree* types	word types	context-free rules
English	6926	131,397	49,206	1524
Chinese	1140	21,125	10,772	515
Korean	632	13,941	10,035	152

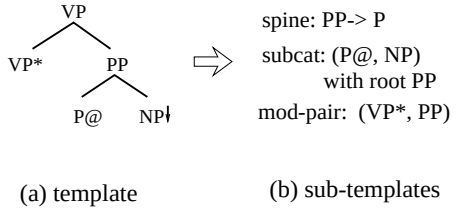
Table 2: Grammars extracted from three Treebanks

4 Comparing Treebank Grammars for Different Languages

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



(a) template (b) sub-templates

Figure 5: The decomposition of an *etree* template (In sub-templates, @ marks the anchor in subcategorization frame, * marks the modifiee in a modifier-modifiee pair.)

that, we first create a new tagset that includes all the tags from the three Treebanks. Then we merge several tags in this new tagset into a single tag.[3] 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, context-free rules, 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;[4] Second, we read context-free rules off the templates and compare these context-free rules; Third, we decompose each template into a list of *sub-templates* (e.g., spines and subcategorization frames) and compare these sub-templates. 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 5 shows the decomposition of a mod-etree template.

4.2 Initial Results

After the tags in original Treebanks have been 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 context-free rules and sub-templates(e.g., spines and subcategorization frames) in each grammar.

The third column of 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 that are covered by these common template types. For example, there

[3] 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 past tense verbs, 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.

[4] 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 currently building.

	templates	context-free rules	*subtemplates*				
			spines	subcat frames	mod-pairs	conj-tuples	total
Eng	3139	754	500	541	332	53	1426
Ch	547	290	108	180	152	18	458
Kor	256	102	43	65	54	5	167

Table 3: Treebank grammars with the new tagset

		templates	context-free rules	sub-templates
(Eng, Ch)	type (#)	237	154	246
	token (%)	80.1/81.5	88.0/85.2	91.4/85.2
(Eng, Kor)	type (#)	54	61	96
	token (%)	47.6/85.6	53.4/92.2	58.9/98.4
(Ch, Kor)	type (#)	43	44	69
	token (%)	55.9/81.0	63.2/89.3	65.7/96.0

Table 4: Comparisons of templates, context-free rules, and sub-templates in three Treebank grammars

are 237 template types that appear in both English and Chinese Treebank grammars. These 237 template types account for 80.1% of the template tokens in the English Treebank, and 81.5% of the template tokens in the Chinese Treebank. The table shows that, although the numbers of matched templates are not very high, most of these templates have high frequency and therefore account for the majority of the 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 the template tokens in the English Treebank.

If we compare sub-templates, rather than templates, the percentages of matched sub-template tokens (as shown in the last column in Table 4) are higher than the percentages of matched template tokens. This is because two distinct templates may have common sub-templates. Similarly, the percentages of matched context-free rules (see the fourth column in Table 4) are higher than the percentages of matched template tokens.

4.3 Results Using Thresholds

The comparison results shown in Table 4 used every template in the Treebank grammars regardless of the frequency of the template in the corresponding Treebank. One potential problem with this approach is that some annotation errors in the Treebanks could have a substantial effect on the comparison results. One such scenario is as follows: To compare languages A and B, we use Treebanks T_A for language A and Treebank T_B for language B. Let G_A and G_B be the grammars extracted from T_A and T_B, respectively, and let t be a template that appears in both grammars. Now suppose that t is a linguistically valid template for language A and it accounts for 10% of the template tokens in T_A, but t is not a valid template for language B and it appears once in Treebank B only due to annotation errors. In this scenario, if G_B *excluding* template t covers 50% of the template tokens in Treebank A, then G_B *including* t covers 60% of the template tokens in Treebank A. In other words, the single error in Treebank B, which results in template t being included in G_B, changes the comparison results dramatically.

Because most templates that are due to annotation errors occur very infrequently in the Treebanks, we used a threshold to discard from the Treebanks and Treebank grammars all the templates with low frequency in order to reduce the effect of Treebank annotation errors on the comparison results. Table 5 shows the numbers of templates in the Treebank grammars when the threshold is set to various values. For example, the last column lists the numbers of templates that occur at least 40 times in the Treebanks.

Table 6 shows the numbers of matched templates and the percentages of matched template tokens when the low frequency templates are removed from the Treebanks and Treebank grammars. As the value of the threshold increases, for each language pair the number of matched templates decreases. The percentage of matched template tokens might decrease a fair amount at the beginning, but it levels off after the threshold reaches a certain value. This tendency is further illustrated in Figure 6. In this figure, the X-axis is the threshold value, which ranges from 1 to 40; the Y-axis is the percentage of matched template tokens in each Treebank when the templates with low frequency are discarded. The curve on the top is the percentage of template tokens in the Chinese Treebank that are covered by the English grammar, and the curve on the bottom is the percentage of template tokens in the English Treebank that are covered by the Chinese grammar. Both curves become almost flat once the threshold value reaches 6 or larger. This result implies that most templates due to annotation errors occur less than six times in the Treebanks.

To summarize, in order to get a better estimate of the percentage of matched template tokens, we should disregard the low frequency templates in the Treebanks. We have shown that this strategy reduces the effect of annotation errors on the comparison results (see Table 6 and Figure 6). This strategy also makes the difference between the sizes of our three Treebanks less important because once a Treebank reaches certain size, the new templates extracted from additional data tend to have very low frequency in the whole Treebank.

	1	2	3	4	5	10	20	30	40
English	3139	1804	1409	1209	1065	762	524	444	386
Chinese	547	341	272	226	210	155	122	110	100
Korean	256	181	146	132	122	94	67	57	53

Table 5: The numbers of templates in the Treebank grammars with the threshold set to various values

	threshold	1	2	3	4	5	10	20	30	40
(Eng, Ch)	type (#)	237	165	128	111	100	73	54	47	42
	token (%)	80.1/81.5	76.5/80.8	64.3/80.6	64.1/80.6	63.8/78.5	58.1/77.9	57.3/76.9	57.3/76.0	56.4/76.0
(Eng, Kor)	type (#)	54	47	37	35	32	29	23	21	19
	token (%)	47.6/85.6	47.5/81.0	47.2/81.0	47.3/80.7	47.3/80.7	47.3/80.4	47.5/79.3	47.5/79.4	47.6/79.3
(Ch, Kor)	type (#)	43	36	34	29	27	22	18	18	17
	token (%)	55.9/81.0	56.0/81.0	56.0/77.0	56.1/76.0	55.8/76.1	56.0/74.3	56.1/74.5	56.3/74.9	56.5/74.9

Table 6: Matched templates in the Treebank grammars with various threshold values

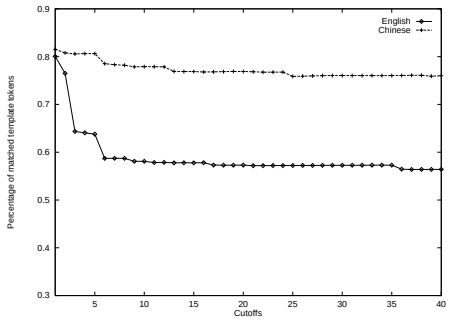


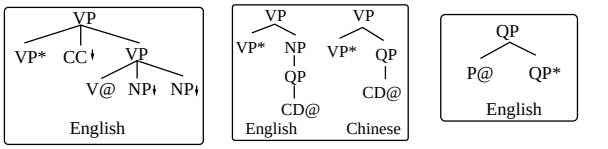
Figure 6: The percentages of matched template tokens in the English and Chinese Treebanks with various threshold values

4.4 Unmatched Templates

Our experiments (see Table 4 and 6) show that the percentages of unmatched template tokens in three Treebanks range from 14.4% to 52.4%, depending on the language pairs and the threshold value. Given a language pair, there are various 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 those 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 had been complete, uniformly annotated, and error-free. A spuriously unmatched template might exist because of one of the following reasons:

(S1) Treebank coverage: 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.



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

Figure 7: Spuriously unmatched templates

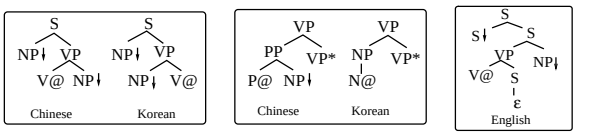
(S2) Annotation differences: Treebanks may choose different annotations for the same constructions; consequentially, the templates for those constructions look different. Figure 7(S2) shows the templates used in English and Chinese for a VP such as "*surged 7 (dollars)*". In the template for English,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

(S3) Treebank annotation errors: 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 adverb *about* in the sentence *About 50 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 might exist.

(T1) Word order: The word order determines the positions of dependents with respect to their heads. If two languages have different word orders, the templates that include dependents of a head are likely to look different. For example, Figure 8(T1) shows the templates for transitive verbs in Chinese and Korean grammars. They 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



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

Figure 8: Truly unmatched templates

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 nouns), as shown in the right figure in Figure 8(T2); therefore, the templates with PP in Chinese, 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.

	S1	S2	S3	T1	T2	T3	total
type(#)	1	70	53	22	99	65	310
token(%)	0.0	3.2	0.2	0.7	12.3	2.1	18.5

Table 7: The distribution of the Chinese templates that do not match any English templates

So far, we have listed six possible reasons for unmatched templates. We have manually classified templates that appear in the Chinese grammar, but not in the English grammar.[5] The results are shown in Table 7. The table shows that for the Chinese-English pair, the main reason for unmatched templates is (T2); that is, the Chinese Treebank has tags for particles (such as aspect markers and sentence-ending particles), which do not exist in English. For other language pairs, the distribution of unmatched templates may be very different. For instance, Table 4 indicates that the English grammar covers 85.6% of the template tokens in the Korean Treebank. If we ignore the word order in the templates, that percentage increases from 85.6% to 97.2%. In other words, the majority of the template tokens that appear in the Korean Treebank, but not in the English Treebank, are due to the word order difference in the two languages. Note that the word order difference only accounts for a small fraction of the unmatched templates in the Chinese-English pair (see the fifth column in Table 7). This contrast is not surprising considering that English and Chinese are predominantly head-initial, whereas Korean is head-final.

5 Conclusion

We have presented a method of quantitatively comparing grammars extracted from Treebanks. Our experimental results show a high proportion of easily inter-mappable structures, providing support for the Universal Grammar hypothesis. We have also described a number of reasons why a particular template does not match any templates in the 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 would 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] can be automatically detected.

[5] For this experiment, we used all the templates in the grammars; that is, we did not throw away low frequency templates.

References

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

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

[Han *et al.*, 2001] Chunghye Han, Na-Rae Han, and Eon-Suk Ko. Bracketing Guidelines for the Penn Korean Treebank (forthcoming), 2001. www.cis.upenn.edu/~xtag/koreantag.

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

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

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

[Sarkar, 2001] Anoop Sarkar. Applying Co-Training Methods to Statistical Parsing. In *Proc. of the 2nd NAACL*, 2001.

[Xia *et al.*, 2000a] Fei Xia, Martha Palmer, and Aravind Joshi. 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)*, 2000.

[Xia *et al.*, 2000b] Fei Xia, Martha Palmer, Nianwen Xue, Mary Ellen Okurowski, John Kovarik, Shizhe Huang, Tony Kroch, and Mitch Marcus. 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, 2000.

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