Natural Language Engineering **1** (1): 1–15. Printed in the United Kingdom

Bootstrapping Parsers via Syntactic Projection across Parallel Texts

Rebecca Hwa* Philip Resnik[§,¶] Amy Weinberg[§,¶] Clara Cabezas[§,¶] Okan Kolak[§,‡]

*Department of Computer Science, Univ. of Pittsburgh, PA 15260
hwa@cs.pitt.edu
§Institute for Advanced Computer Studies, Univ. of Maryland, College Park, MD USA 20742
¶Department of Linguistics, Univ. of Maryland, College Park, MD USA 20742
‡Department of Computer Science, Univ. of Maryland, College Park, MD USA 20742
{resnik,weinberg,clarac,okan}@umiacs.umd.edu †

(*Received 30 April 2004; revised 30 November 2004*)

Abstract

Broad coverage, high quality parsers are available for only a handful of languages. A prerequisite for developing broad coverage parsers for more languages is the annotation of text with the desired linguistic representations (also known as "treebanking"). However, syntactic annotation is a labor intensive and time-consuming process, and it is difficult to find linguistically annotated text in sufficient quantities. In this article, we explore using parallel text to help solving the problem of creating syntactic annotation in more languages. The central idea is to annotate the English side of a parallel corpus, project the analysis to the second language, and then train a stochastic analyzer on the resulting noisy annotations. We discuss our background assumptions, describe an initial study on the "projectability" of syntactic relations, and then present two experiments in which stochastic parsers are developed with minimal human intervention via projection from English.

1 Introduction

More and more frequently, researchers and developers in natural language processing are confronted with a need to develop language technology components in new languages. In light of the recent success of corpus-based approaches, the logical first question to be asked is, "Where can I get some annotated data?" For many kinds of linguistic problems, having annotated data in hand means being able to get off the ground — which helps explain why the Linguistics Data Consortium's list of "tools and formats for creating and managing linguistic annotations" has 58 entries.[1]

† The authors gratefully acknowledge helpful discussions with Adam Lopez and Gina Levow, the constructive comments of the anonymous reviewers, as well as publicly available software used in this work. This research was supported in part by National Science Foundation grant EIA0130422, Department of Defense contract RD-02-5700, and ONR MURI Contract FCPO.810548265.

[1] The list is maintained at http://www.ldc.upenn.edu/annotation/.

One of NLP's long standing central problems, syntactic parsing, illustrates the importance of annotated resources. From the empirical studies of several state-of-the-art statistical parsers, the common experience is that when training on manually annotated Treebank data of different languages (Ratnaparkhi 1999; Bikel and Chiang 2000), larger treebanks ensure better performances. Hidden behind the numbers is the reality of what it takes to obtain the data: the 4,000 parse trees in Penn Chinese Treebank Version 2 first appeared two years after the start of the project, and the increase to Version 4 appeared three years after the release of Version 2.

One actively researched approach to this problem is to develop weakly supervised algorithms that require less training data, such as active learning (Hermjakob and Mooney 1997; Tang et al. 2002; Baldridge and Osborne 2003; Hwa 2004) and co-training (Sarkar 2001; Steedman et al. 2003). In this article, we explore an alternative: using parallel text as a means for transferring syntactic knowledge from a resource-rich language to a language with fewer resources. The central idea is to annotate the English side of a parallel corpus, project the analysis to the second language, and then train a statistical parser on the resulting noisy annotations. Annotation projection using parallel text has been accomplished for shallower tasks(Yarowsky and Ngai 2001; Merlo et al. 2002; Yarowsky et al. 2001), but the projection of tree structures introduces additional complexity. The success of the approach hinges on two questions. First, is it possible to infer complex structures for a second language from monolingual representations in English, with a minimum of human intervention? And second, since automatic projection of dependencies leads to noisy representations, can larger quantity offset lower quality when training a stochastic parser?

Section 2 takes up the first of these questions, motivating the syntactic representations with which we are concerned and the prospects for using English as a basis for inferring representations in a second language. Section 3 describes an initial study considering the second question, developing and evaluating a stochastic parser for Spanish. Section 4 tackles a more realistic scenario involving the acquisition of a parser for Chinese, a language that is less similar to English. Section 5 concludes with a summary and directions for future work.

2 Projection of Syntactic Dependencies

We follow Lin (1998) in adopting dependency-based representations for the work described in this article. Dependency representations have a number of desirable properties that make them useful in NLP applications. First, they allow us to characterize long-distance syntactic relationships between words. Even in the area of stochastic language modeling (where "shallow" string-based approaches such as n-gram models have dominated for decades), there is recent evidence suggesting the value of syntactic structure (Chelba et al. 1997; Chelba and Jelinek 1998; Charniak 2001; Khudanpur and Wu 2000). Consider the following example from the Brown Corpus:

The largest hurdle the Republicans would have to face is a state law which says that before making a first race, one of two alternative courses must be taken.

The relationship between *hurdle* and *is* exists over a long string-distance, owing to an embedded relative clause; similarly, *Republicans* and *face* are separated in the string by a sequence of auxiliaries and the infinitival *to*. As a result, the relationships represented in the sentence are not captured well by any n-gram model with tractable n. In contrast, the relationship between the subject NP and the predicate is easily encoded locally if one can represent the relationships between phrases rather than just among contiguous sequences of words.

Second, it is widely recognized that phrase structure representations provide an implicit representation of the syntactic dependency relationships between words in the structure — that is, asymmetric binary relations between *heads* and *dependents*, capturing such grammatical relations as 'subject', 'object', 'modifier', and the like. Lin observes that syntactic dependencies, more than syntactic constituents, are closely tied to the who-did-what-to-whom relationships of language. Since semantic dependencies form a superset based on syntactic dependencies, measuring correctness of dependencies rather than constituents is more likely to reflect how likely a representation is to be interpretable.[2] Moreover, Lin notes that in the parser evaluation literature, standard measures based on phrase structure constituency usually compare the phrase boundaries specified by the phrase structure grammar of a gold standard test set to those of the candidate analysis. However, because constituents' branching structure is not directly tied to semantic interpretation, it is unclear how to assess the seriousness of missing, spurious, or crossing branches; in contrast, each explicit link in a dependency representation captures a single head-dependent relationship.

Finally, with respect to the projection of dependencies, the process we will describe carries information across languages with varying word orders; therefore it is imperative to firmly separate precedence from the dominance structure that carries semantic information. For example, the relative string order of a series of modifiers of a head is irrelevant in a dependency representation — all are modifiers. By contrast, a constituency tree may require a stacked structure that would not translate well if the word order were reversed in the second language (Fox 2002).

2.1 Projecting Dependencies

The above observations provide our motivation for developing an approach to cross-language inference of syntactic representations using syntactic dependencies, rather than syntactic constituents. Having made that choice, the success of syntactic annotation projection depends crucially on the question of whether or not the syntactic dependencies in English sentences can reasonably be assumed to give rise to corresponding syntactic dependencies in their second-language translations. This assumption can be formalized as follows:

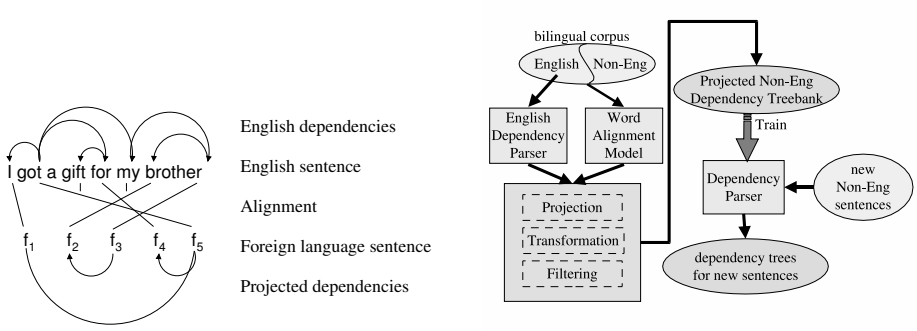
[2] Dependency-based approaches to syntax have a long history in linguistics; see, e.g., (Mel'cuk 1988). The relationship between dependency and phrase-structure grammars has been well studied in the linguistics literature (e.g. Abney(1995)) and localization of lexical dependency constraints has proven useful in context-free parsing for constituency representations (Collins 1997; Charniak 1999).

Given sentence pair (E, F) and a set of syntactic relations for E, where $E = e_1, \ldots, e_n$ is an English sentence and $F = f_1, \ldots, f_m$ is its non-English parallel, syntactic relations (denoted as $R(x, y)$) are projected from English for the following situations:

- **one-to-one** if e_i is aligned with a unique f_x and e_j is aligned with a unique f_y, if $R(e_i, e_j)$,conclude $R(f_x, f_y)$.
- **unaligned (English)** if e_j is not aligned with any word in F, then create a new empty word f_y such that for any e_i aligned with a unique f_x, $R(e_i, e_j) \Rightarrow R(f_x, f_y)$ and $R(e_j, e_i) \Rightarrow R(f_y, f_x)$.
- **one-to-many** if $e_i$is aligned with $f_x, \ldots, f_y$, then create a new empty word f_z such that f_z is the parent of $f_x, \ldots, f_y$ and set e_i to align to f_z instead. We called this a *Multiply-Aligned Component, or (MAC).*
- **many-to-one** if $e_i, \ldots, e_j$ are all uniquely aligned to f_x, then delete all alignments between $e_k (i \leq k \leq j)$ and f_x except for the head of $e_i, \ldots, e_j$.
- **many-to-many** decomposed into a two-step process: first perform one-to-many, then perform many-to-one.

Leave unaligned words in F out of the projected syntactic tree.

Fig. 1. The Direct Projection Algorithm



(a) (b)

Fig. 2. (a) Projecting a dependency tree from an English sentence to a parallel Basque sentence. (b) Our projection system architecture.

Direct Correspondence Assumption (DCA): Given a pair of sentences E and F that are (literal) translations of each other with syntactic structures $Tree_E$ and $Tree_F$, if nodes x_E and y_E of $Tree_E$ are aligned with nodes x_F and y_F of $Tree_F$, respectively, and if syntactic relationship $R(x_E, y_E)$ holds in $Tree_E$, then $R(x_F, y_F)$ holds in $Tree_F$.

As stated, the DCA amounts to an assumption that the cross-language alignment resembles a homomorphism relating the syntactic graph of E to the syntactic graph of F. Whether or not stated explicitly, the DCA is actually an underlying assumption in most formal attempts to model cross-language correspondences in syntactic relationships (Wu 1995; Alshawi et al. 2000; Yamada and Knight 2001; Eisner 2003; Gildea 2003; Melamed et al. 2004; Smith and Smith 2004).

Table 1 illustrates the principle with the following English-Basque sentence pair as an example:

(1) a. I got a gift for my brother

Relation R	Head x_{Eng}	Modifier y_{Eng}	Head x_{Bsq}	Modifier y_{Bsq}
verb-subj	got	I	erosi	nik
verb-obj	got	gift	erosi	opari
noun-det	gift	a	opari	bat
noun-mod	brother	my	anaiari	nire

Table 1. *Correspondences preserved in the English-Basque example.*

b. Nik (I) nire (MY) anaiari (BROTHER-DAT) opari (GIFT) bat (A) erosi (BUY) nion (PAST)

Observe that, given the indicated alignments of words, many of the English sentence's central relations carry over to the Basque sentence, despite the fact that the languages are quite different.

The DCA gives rise to a straightforward projection procedure (Figure 1) in which the dependencies in an English sentence are projected to the sentence's translation, using the word-level alignments as a bridge.[3] Figure 2(a) depicts the resulting projected dependency tree when the Direct Projection Algorithm is applied to our English-Basque example.

To investigate the reasonableness of the DCA, which one might characterize as the "projectability" of English syntactic dependencies, we have conducted two experiments under an idealized setting, using "perfect" (i.e. human-generated) English parses and word alignments. In one study, one hundred English dependency trees are projected onto their parallel Spanish sentences; in the other study, a set of 88 English trees are projected onto their parallel Chinese sentences. The projected non-English trees are then compared with manually generated gold standard trees (the development of the gold standards will be described in fuller detail later in the main experiment section). For both Spanish and Chinese, we found similar results: that the directly projected dependency trees yield many mistakes. For Spanish, the unlabeled dependency F-score is 37%; for Chinese, the unlabeled dependency F-score is 38%.

2.2 Post-Projection Transformation

Although the direct projection algorithm was not very successful by itself, many of its errors did not implicate the DCA itself; rather, they highlighted the fact that second-language parses required *more* than just the projection of the English dependencies — they also required a certain amount of monolingual knowledge specific to the projected-to language. For example, Chinese verbs are often followed by an aspectual marker that is not realized as a word in English; because the Direct

[3] Further details of this experiment can be found in our earlier work (Hwa et al. 2002).

	Direct Projection	Projection + Transformation
English-Spanish	36.8	70.3
English-Chinese	38.1	67.3

Table 2. *An evaluation of dependency structures projected from English under the ideal setting of projecting from manually generated English dependency trees across manually aligned words.*

Projection Algorithm will *only* attach words on the basis of information projected from English, those markers will *always* be left unattached in the Chinese parses. Similarly problems arise with Spanish clitics (*le*, *se*, etc.), which are separated from their verbs after tokenization. For example, a Spanish sentence *Ella va a dormirse* would be tokenized as *Ella va a dormir se*, and projection from the corresponding English sentence (she is going to fall asleep) would leave the word *se* unattached.

Error analysis led us to revise our projection approach to incorporate a small set of *correction rules* to be carried out post-projection. The rules are expressed in a tree-based pattern-action formalism, performing local transformations of a projected analysis on the non-English side. It is clearly an advantage to limit such rules to those that can apply generally, across many construction types. Wishing to avoid unending language-specific rule tweaking, we strictly limited the possible rules, permitting them to refer only to closed class items (such as aspectual markers), to parts of speech projected from the English analysis, or to easily enumerated lexical categories (e.g. pronouns, prepositions, Chinese measure words). Moreover, we focused on rules motivated by general linguistic properties of the language. As an example, the rule handling aspectual markers takes the following form:

- An aspectual marker should modify the verb to its left.

Thus, if $f_x, \ldots, f_y$ is a sequence of Chinese words aligned with an English verb, and it is followed by f_a, an aspect marker, then we make f_a into a modifier of the last verb f_y. See the Appendix for more Chinese rules.

Viewing the problem from a higher level of linguistic abstraction made it possible to find relevant cases in a short time and express the solution compactly: for the idealized study, fewer than dozens of post-projection transformation rules (written within a month in the worst case) captured the bulk of the missing language-specific information. With the application of the transformational rules after the application of the Direct Projection Algorithm, the resulting parses obtained an F-score of 70% for Spanish and 67% for Chinese (Table 2).

Although these results obtained in the idealized scenario — starting with manually generated English parses and word alignments — they demonstrate the promise of developing parsers via projection of syntactic dependency information from English. With only a few weeks of language-specific work, we obtained trees for two non-English languages of a quality that would have been extremely challenging (perhaps impossible) to obtain without the importation of syntactic dependency knowledge from English by way of parallel translation.

2.3 Our Projection Framework for Bootstrapping Parsers

The encouraging results from the pilot studies presented in the previous section suggest that the projected trees, although imperfect, may be good enough to be used as training data to bootstrap a new parser in the non-English language. Figure 2(b) lays out our complete framework. First, we need a sizable, sentence-aligned bilingual text as training corpus. The English side of the parallel text is analyzed by Collins's (1997) Model 2 parser (trained on the Wall Street Journal portion of the Penn Treebank) and then converted to a dependency representation based on a standard head-table approach (Magerman 1994), using an algorithm similar that of Xia and Palmer (2001).[4] The parallel corpus is aligned at the word level using the GIZA++ implementation of the IBM statistical translation models (Brown et al. 1990; Al-Onaizan et al. 1999; Och and Ney 2003). We then project the English dependency structures across the word alignment to the non-English side in accordance with our Direct Projection Algorithm.[5] Next, we apply a small set of language-specific post-projection transformation rules to address some language-specific information. As discussed earlier, the set of post-projection transformation rules is very small and only expresses general linguistic constructs, so it can be developed within a short period of time (see Appendix). Finally, to address the problem of propagating English parsing errors and word-alignment errors, we apply an aggressive filtering strategy to automatically prune out projected trees that we predict to be of poor quality. These filtering criteria are discussed in more detail in the experimental sections. The remaining set of projected trees becomes the treebank that will be used to train a new dependency parser — we conduct our experiments using a version of the Collins parser that has been adapted for dependency treebanks (Collins et al. 1999). Once trained, the new parser is ready to generate dependency analyses for unseen new sentences in that language. In the next two sections, we evaluate the quality of parsers trained in this fashion.

3 Study I: Creating a Spanish Parser

In this section and the section that follows, we demonstrate the promise of the approach in progressively more challenging scenarios. The first of these scenarios is projection from English to Spanish in order to create a Spanish parser.

3.1 Experimental Setup

For this experiment, we used a parallel corpus of 100,000 English-Spanish sentences, constructed by combining verses from modern-day Bible translations, a sample from the Federal Broadcast Information Service (FBIS) English-Spanish corpus, and a

[4] The Collins (1997) Model 2 parser can be downloaded at ftp://ftp.cis.upenn.edu/pub/mcollins/misc.

[5] Since the IBM models do not produce many-to-many word alignments, the Direct Projection Algorithm's rules pertaining to many-to-many alignments are not activated.

sample from the United Nations Parallel Corpus. A set of 200 sentence pairs was excluded from the training set to be used as development and test data (100 sentences each). Because no Spanish treebank was available, we created a gold standard in the following manner. Each sentence was first parsed automatically using a state-of-the-art commercial Spanish parser based on constraint grammar (Karlsson 1990), which produces dependency-like representations. We then asked two linguists with extensive training in Spanish syntax to correct the parser's output, e.g. writing a parse tree from scratch in cases where it failed to create a parse, adding missing dependencies for unattached words, or correcting head-dependent relationships. The annotators were also asked to make stylistic adjustments that represented standard dependency assumptions in the treebanking community — for example, we asked them to represent the preposition rather than its object as the head of a prepositional phrase (Xia and Palmer 2001; Zabokrtsky and Smrz 2003). The two linguists worked independently of each other, and were not involved in any other aspect of this experiment. Their annotations agree with each other with an average unlabeled F-score of 84.7% over the development and test sentences.

3.2 Evaluation

Before evaluating the bootstrapped parser directly, we first repeated the study performed in Section 2, this time under more realistic conditions using automatically generated English parse trees and word alignments. Table 3 shows that, as expected, with the introduction of additional errors from the English parser and the word alignment model, the projected trees have more errors.[6] However, the degree of degradation is relatively low (a drop of 5% in F-score). For a baseline, we generated dependency trees in which every word modifies the word to its left (respecting the basic word order of the language, since Spanish is head initial). As a point of comparison, we also passed the baseline trees through post-projection transformation. If the transformation rules are too specific (such that they essentially constitute a rule-based parser), they will help the baseline trees as much as the projected trees. The baseline results show that this is not the case: the fact that the automatically projected trees have much better performance than the baseline suggests that lexical and syntactic knowledge has indeed been transferred over from the English side to the Spanish side.

To decrease the chance of adding poor quality dependency trees into the projected treebank, we employed the following pruning criteria (based on tuning on development set):

- Discard if more than 20% of the English words have no Spanish counterpart.
- Discard if more than 30% of the Spanish words have no English counterpart.
- Discard if more than 4 Spanish words were aligned to the same English word.

[6] The alignment word error rate is 24.4%. We did not measure the English parser error rate for this data; by inspection, we believe the output quality is comparable to that of the standard PennTreebank test data, thus we expect the error rate to be 12-15%.

	Direct Projection	Projection + Transformation
Baseline (mod prev)	31.0	39.1
Automatic	33.9	65.7
Manual (Ideal)	36.8	70.3

Table 3. *An evaluation of dependency structures projected to Spanish from the output of off-the-shelf softwares (automatic), compared against a baseline and an upper bound — the ideal setting (from Table 2).*

Method	Corpus	Train Size	Parsing Performance
Baseline (mod prev)	–	–	33.8%
Stat. parser	UN/FBIS/Bible (no filter)	98K sents	67.3%
Stat. parser	UN/FBIS/Bible (w/ filter)	20K sents	72.1%
Commercial parser	–	–	69.2%

Table 4. *A comparison of parsing performances of different approaches (measured against two independent annotators)*

Without filtering, the projection process results in a set of 98,000 projected dependency trees; with filtering, the number of trees reduces to 20,000.

Table 4 summarizes the experimental results of training a Spanish parser from the noisy projected treebank. We compared the bootstrapped parsers (one trained on the filtered corpus and one on the unfiltered corpus) and the commercial parser's output.[7] The parsers are evaluated on their parsing performance on the test sentences, using the unlabeled F-score as a metric. The table shows that with the help of linguistically informed effort over a short period of time, projection of syntactic dependencies across a parallel corpus yields performance that is comparable with a state of the art rule-based commercial system that presumably took considerably longer to construct.

4 Study II: Creating a Chinese Parser

Although the results from Section 3 are encouraging, it is important to find out how the idea will hold up under more stringent conditions. Spanish and English are similar enough that automatic word alignment methods can be expected to perform reasonably well; and in general their syntactic structures can be expected

[7] The scores were computed after applying a set of (reversible) deterministic transformational rules on the commercial parser's outputs to minimize purely stylistic differences.

to be somewhat similar compared to other language pairs. In addition, although we made every effort to create a fair test set, it is small, and not an *independently constructed* test set; therefore, there is always the possibility of unintended bias. To address these issues, we now consider the more challenging task of bootstrapping a Chinese parser using projection across English-Chinese parallel text.

4.1 Experimental Setup

The parallel corpus used for training in this experiment consists of 240,000 sentence pairs from FBIS.[8] Automatic word alignment for English-Chinese is much more difficult than English-Spanish. Comparing against the manually aligned gold standard, we estimate the word alignment error rate to be 41%, as contrasted with the English-Spanish alignment error rate of 24.4%.

For evaluating the projected Chinese parser, we derived the gold-standard parses in our development and test set from the Penn Chinese Treebank Version 2. This was done by automatically converting the Treebank's constituency parses of the Chinese sentences into syntactic dependency representations, using the same constituency-to-dependency algorithm we used for English trees in Section 2.3.[9] We reserved a small subset of the Treebank data for development purposes (sentences from sections 001-015, 038, 039, 067, 122, 191, 207, 249). The remaining sentences whose lengths are 40 words or less were used as test data, resulting in a large test set of about 2800 sentences. The average sentence length of the test set is 20.6 words.

4.2 Evaluation

As in the English-Spanish study, we first evaluate the quality of the projected Chinese dependency trees — repeating the pilot study in Section 2, but under more realistic conditions. The results are summarized in Table 5. This study echoes our earlier findings for Spanish. As before, the baseline performs poorly, even after post-projection transformation.[10] The degradation in the quality of the projected trees from using automatic methods is more pronounced. This is due to both the dissimilarity of the language pair and the increase in the number of word alignment errors.

Next, we evaluate the parser trained from the projected treebank. Table 6 compares the results. Because English-Chinese word alignments are more prone to error, filtering out badly aligned sentences is even more important. In addition to the criteria mentioned earlier, we also factored in criteria such as the number of

[8] In keeping with common practice in statistical machine translation, to improve the quality of word alignment, we appended an English-Chinese word list (LDC) to the parallel text during (and only during) alignment.

[9] The strategy was validated by a human linguist performing the same process on the development data set; the agreement rate with the human-generated dependency trees was 97.5%. This led us to be confident that Treebank constituency parses could be used automatically to create a gold standard for syntactic dependencies.

[10] Because Chinese is head-final, the baseline is for each word to modify the word to its immediate right.

	Direct Projection	Projection + Transformation
Baseline (mod next)	35.9	43.1
Automatic	26.3	52.4
Manual (ideal)	38.1	67.3

Table 5. *An evaluation of dependency structures projected to Chinese from the output of off-the-shelf softwares (automatic), compared against a baseline and an upper bound — the ideal setting (from Table 2).*

Method	Corpus	Train Size	Parsing Performance
Baseline (mod next)	–	–	35.1%
Baseline + transformations	–	–	44.3%
Stat. parser	FBIS (w/ filter)	50K sents	53.9%
Stat. parser	ChTB (new in v4)	10K sents	64.3%

Table 6. *A comparison of parsing performances of different approaches (measured against unseen test set taken from ChTB v2.)*

cross-dependency links, the number of nodes remaining unattached even after the post-projection transformation, and the number of words that could not be part-of-speech tagged. Out of the 240,000 sentence pairs, only 50,000 sentences remained after filtering.[11] The bootstrapped parser's performance is squarely between the baseline (modify-next plus transformations) and the upper bound obtained by training on noise-free dependency structures derived from the Penn Chinese Treebank Version 4.

To determine how the bootstrapped parser compares with a parser trained on human annotated data, we analyze the upper-bound parser by examining its learning curve. Figure 3 plots the learning curve of a parser trained on manually annotated data. The more labeled training sentences the parser receives (x-axis) the better its performance on test data (y-axis). We see that the bootstrapped parser has the same level of parsing performance as a parser trained on over 2000 noise-free dependency trees.

[11] In a follow up experiment, we found that, assuming equivalent word alignment quality, reducing the parallel corpus to one-fifth of the full size (such that after filtering, 10,000 sentences remained) resulted in an absolute degradation of only 1.4% in parsing performance. This suggests that the technique is also likely to be robust for smaller corpora.

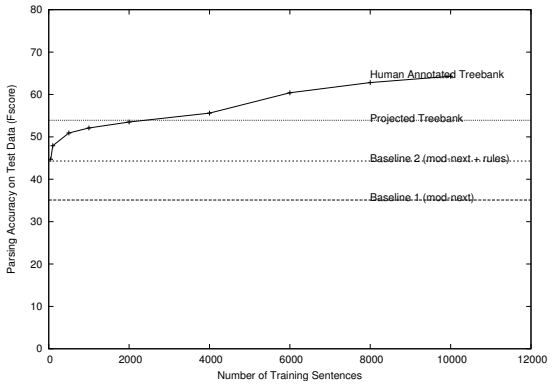


Fig. 3. Learning curve of a parser trained on manually annotated treebank.

5 Conclusions

It is not easy to build treebanks that support training of stochastic parsers. Abeillé (2003) presents an informative picture of the state of the art for treebank annotation. From evidence across a range of projects, it appears that acquiring 20,000-40,000 sentences — including the work of building style guides, redundant manual annotation for quality checking, and so forth — can take from four to seven years. The research presented here constitutes a successful first step in applying the annotation projection approach to syntactic representations. In contrast to the focus in research on tree-to-tree or bilingual grammar models (Wu 1997; Eisner 2003), we produce not a model but a treebank, which can be used for stochastic parser training or as the starting point for manual correction (Marcus et al. 1993).

As Figure 3 illustrates, even for a difficult case like Chinese, the annotation projection approach produces trees that contain enough information for a stochastic parser to get past the initial steep climb up the learning curve. Our experiments showed that the parser performance from an automatically projected Chinese treebank is only a few points below what one would obtain after one or two years of manual treebanking, while requiring less than one person-month writing manual correction rules to account for limitations in projecting dependencies from English.

The process of performing this research has exposed the importance of distinguishing what can be projected versus what can only be learned on the basis of monolingual information in the language to be parsed. In current research, we are exploring the possibility of starting with a small, manually produced seed corpus in order to provide the key monolingual facts, and iteratively improving that corpus using information projected from English. For example, in the English-Chinese case, the trees projected from English may make it possible to confidently identify many of the verb-argument relations, and a small number of confidently annotated Chinese trees may suffice to teach the parser how to identify attachment points for aspectual markers.

References

Anne Abeillé, editor. 2003. *Treebanks: Building and Using Parsed Corpora*. Kluwer.

Steven Abney. 1995. Dependency grammars and context-free grammars. Presented at meeting of Linguistic Society of America, January.

Yaser Al-Onaizan, Jan Curin, Michael Jahr, Kevin Knight, John Lafferty, I. Dan Melamed, Franz-Josef Och, David Purdy, Noah A. Smith, and David Yarowsky. 1999. Statistical machine translation. Technical report, JHU. http://citeseer.nj.nec.com/al-onaizan99statistical.html.

Hiyan Alshawi, Srinivas Bangalore, and Shona Douglas. 2000. Learning dependency transduction models as collections of finite state head transducers. *Computational Linguistics*, 26(1).

Jason Baldridge and Miles Osborne. 2003. Active learning for HPSG parse selection. In *Proceedings of the 7th Conference on Natural Language Learning*, Edmonton, Canada, June.

Daniel M. Bikel and David Chiang. 2000. Two statistical parsing models applied to the chinese treebank. In *Proceedings of the Second Chinese Language Processing Workshop*, pages 1–6, Hong Kong.

Peter F. Brown, John Cocke, Stephen A. DellaPietra, Vincent J. DellaPietra, Frederick Jelinek, John D. Lafferty, Robert L. Mercer, and Paul S. Roossin. 1990. A statistical approach to machine translation. *Computational Linguistics*, 16(2):79–85, June.

Eugene Charniak. 1999. A maximum-entropy inspired parser. Technical Report CS-99-12, Brown University.

Eugene Charniak. 2001. Immediate-head parsing for language models. In *Proc. of the 39th Meeting of the Association for Computational Linguistics*, Toulouse, France.

Ciprian Chelba and Fredrick Jelinek. 1998. Exploiting syntactic structure for language modeling. In *Proceedings of the 36th Annual Meeting of the Association for Computational Linguistics and 17th International Conference on Computational Linguistics*, volume 1, pages 225–231, Montreal, Canada.

Ciprian Chelba, David Engle, Frederick Jelinek, Victor M. Jimenez, Sanjeev Khudanpur, Lidia Mangu, Harry Printz, Eric Ristad, Ronald Rosenfeld, Andreas Stolcke, and Dekai Wu. 1997. Structure and performance of a dependency language model. In *Proc. Eurospeech '97*, pages 2775–2778, Rhodes, Greece.

Michael Collins, Jan Hajic, Lance Ramshaw, and Christoph Tillmann. 1999. A statistical parser for czech. In *Proceedings of the 37th Annual Meeting of the Association for Computational Linguistics*, College Park, Maryland.

Michael Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proceedings of the 35th Annual Meeting of the Association for Computational Linguistics*, pages 16–23, Madrid, Spain.

Jason Eisner. 2003. Learning non-isomorphic tree mappings for machine translation. In *Proceedings of the 41st Annual Meeting of the Association for Computational Linguistics (Companion Volume)*, Sapporo, Japan, July.

Heidi Fox. 2002. Phrasal cohesion and statistical machine translation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 304–311, Philadelphia, June.

Daniel Gildea. 2003. Loosely tree-based alignment for machine translation. In *Proceedings of the 41th Annual Conference of the Association for Computational Linguistics (ACL-03)*, Sapporo, Japan.

Ulf Hermjakob and Raymond J. Mooney. 1997. Learning parse and translation decisions from examples with rich context. In *Proceedings of the Association for Computational Linguistics*, pages 482–489, Madrid, Spain.

Rebecca Hwa, Philip S. Resnik, Amy Weinberg, and Okan Kolak. 2002. Evaluating

translational correspondence using annotation projection. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*.

Rebecca Hwa. 2004. Sample selection for statistical parsing. *Computational Linguistics*, 30(3):253–276.

Fred Karlsson. 1990. Constraint grammar as a framework for parsing running text. In Hans Karlgren, editor, *Proceedings of the 13th International Conference on Computational Linguistics*, volume 3, pages 168–173, Helsinki, Finland, August.

S. Khudanpur and J. Wu. 2000. Maximum entropy techniques for exploiting syntactic, semantic and collocational dependencies in language modeling. *Computer Speech and Language*, 14(4):355–372.

Dekang Lin. 1998. Dependency-Based Evaluation of MINIPAR. In *Proceedings of the Workshop on the Evaluation of Parsing Systems, First International Conference on Language Resources and Evaluation*, Granada, Spain, May.

David Magerman. 1994. *Natural Language Parsing as Statistical Pattern Recognition*. Ph.D. thesis, Stanford University University, February.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.

I. Dan Melamed, Giorgio Satta, and Ben Wellington. 2004. Generalized multitext grammars. In *Proceedings of the 42nd Annual Conference of the Association for Computational Linguistics (ACL-04)*, Barcelona, Spain.

Igor A. Mel'cuk. 1988. *Dependency Syntax: Theory and Practice*. The SUNY Press, Albany, NY.

Paolo Merlo, Suzanne Stevenson, Vivianne Tsang, and Gianluca Allaria. 2002. A multilingual paradigm for automatic verb classification. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics (ACL-2002)*, pages 207–214, Philadelphia, Pennsylvania, July.

Franz Josef Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Computational Linguistics*, 29(1):19–51.

Adwait Ratnaparkhi. 1999. Learning to parse natural language with maximum entropy models. *Machine Learning*, 34(1-3):151–175.

Anoop Sarkar. 2001. Applying co-training methods to statistical parsing. In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics*, pages 175–182, Pittsburgh, PA, June.

David A. Smith and Noah A. Smith. 2004. Bilingual parsing with factored estimation: Using english to parse korean. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*.

Mark Steedman, Miles Osborne, Anoop Sarkar, Stephen Clark, Rebecca Hwa, Julia Hockenmaier, Paul Ruhlen, Steven Baker, and Jeremiah Crim. 2003. Bootstrapping statistical parsers from small datasets. In *The Proceedings of the Tenth Conference of the European Chapter of the Association for Computational Linguistics*, pages 331–338, Budapest, Hungary.

Min Tang, Xiaoqiang Luo, and Salim Roukos. 2002. Active learning for statistical natural language parsing. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 120–127, Philadelphia, PA, July.

Dekai Wu. 1995. Stochastic inversion transduction grammars, with application to segmentation, bracketing, and alignment of parallel corpora. In *Proc. of the 14th Intl. Joint Conf. on Artificial Intelligence*, pages 1328–1335, Montreal, Aug.

Dekai Wu. 1997. Stochastic inversion transduction grammars and bilingual parsing of parallel corpora. *Computational Linguistics*, 23(3):377–401.

Fei Xia and Martha Palmer. 2001. Converting dependency structures to phrase structures. In *Proc. of the HLT Conference*, March.

Kenji Yamada and Kevin Knight. 2001. A syntax-based statistical translation model. In

Proc. of the Conference of the Association for Computational Linguistics, pages 523–529, Toulouse, France.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Proceedings of the Second Meeting of the North American Association for Computational Linguistics*, pages 200–207.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of HLT 2001, First International Conference on Human Language Technology Research*.

Zdenek Zabokrtsky and Otakar Smrz. 2003. Arabic syntactic trees: from constituency to dependency. In *The Proceedings of the Tenth Conference of the European Chapter of the Association for Computational Linguistics*, Budapest, Hungary. http://ufal.mff.cuni.cz/publications/year2003/eacl-trees.pdf.

Appendix: Post-Projection Rules for Chinese

Post-projection rules can be categorized into two types: those that connect the unattached nodes (because they have no English equivalent) to the projected dependency structure, and those that modify malformed dependency structure (due to many-to-one mapping and other divergences discussed in Section 2.1). To reduce repetitions in the list of rules below, we only highlight the guiding linguistic constraints rather than specific instantiations of the rules.

- The word preceding the token *di* should be labeled as an adverb, and modifies *di*, and *di* modifies the verb to its right.
- An adverb should modify a verb (unless it is modifying *di*, and it takes no modifier itself).
- An aspectual marker should modify the verb to its left.
- The verbs *have* and *be* must have an object modifier.
- Numbers (both ordinal and cardinal) and determiners should either modify a noun to its right or a verb to its left (because the head nouns are sometimes topicalized).
- A measure word modifies a number of a determiner preceding it. It can take no modifer of its own.
- A conjunction needs two or more coordinating words of the same tag type. If a conjunction has exactly two modifiers but they do not coordinate, assume it was the result of a tagging error.
- The token *de* should modify a noun to its right. If it is modified by a noun to its left, then it is acting as a possessive marker, otherwise it is used as a relative clause marker.
- The token *etc* signals an apposition. It bridges between a (usually implicitly) coordinated list of phrase to its left and a base noun phrase to its right.
- A preposition must have an object modifier, which is typically a noun. If a location preposition co-occurs with a location marker, the location marker is the object modifier for the preposition; the location marker requires an object modifier to its left (i.e., between the preposition and the location marker).
- Some prepositions appear in pairs (such as *from ... to ...*). In these cases, the first preposition should modify the second.
- Default – Chinese is head final, so in any unresolved Multiply-Aligned Components, the right-most word should be the head.