Parser Adaptation and Projection with Quasi-Synchronous Grammar Features*

David A. Smith
Department of Computer Science
University of Massachusetts Amherst
Amherst, MA 01003, USA
dasmith@cs.umass.edu

Jason Eisner
Department of Computer Science
Johns Hopkins University
Baltimore, MD 21218, USA
jason@cs.jhu.edu

Abstract

We connect two scenarios in structured learning: **adapting** a parser trained on one corpus to another annotation style, and **projecting** syntactic annotations from one language to another. We propose **quasi-synchronous grammar** (QG) features for these structured learning tasks. That is, we score a aligned pair of source and target trees based on local features of the trees and the alignment. Our quasi-synchronous model assigns positive probability to any alignment of any trees, in contrast to a synchronous grammar, which would insist on some form of structural parallelism.

In monolingual dependency parser adaptation, we achieve high accuracy in translating among multiple annotation styles for the same sentence. On the more difficult problem of cross-lingual parser projection, we learn a dependency parser for a target language by using bilingual text, an English parser, and automatic word alignments. Our experiments show that unsupervised QG projection improves on parses trained using only high-precision projected annotations and far outperforms, by more than 35% absolute dependency accuracy, learning an unsupervised parser from raw target-language text alone. When a few target-language parse trees are available, projection gives a boost equivalent to doubling the number of target-language trees.

*The first author would like to thank the Center for Intelligent Information Retrieval at UMass Amherst. We would also like to thank Noah Smith and Rebecca Hwa for helpful discussions and the anonymous reviewers for their suggestions for improving the paper.

1 Introduction

1.1 Parser Adaptation

Consider the problem of learning a dependency parser, which must produce a directed tree whose vertices are the words of a given sentence. There are many differing conventions for representing syntactic relations in dependency trees. Say that we wish to output parses in the Prague style and so have annotated a small **target corpus**—e.g., 100 sentences—with those conventions. A parser trained on those hundred sentences will achieve mediocre dependency accuracy (the proportion of words that attach to their correct parent).

But what if we also had a large number of trees in the CoNLL style (the **source corpus**)? Ideally they should help train our parser. But unfortunately, a parser that learned to produce perfect CoNLL-style trees would, for example, get both links "wrong" when its coordination constructions were evaluated against a Prague-style gold standard (Figure 1).

If it were just a matter of this one construction, the obvious solution would be to write a few rules by hand to transform the large source training corpus into the target style. Suppose, however, that there were many more ways that our corpora differed. Then we would like to *learn a statistical model to transform one style of tree into another.*

We may not possess hand-annotated training data for this tree-to-tree transformation task. That would require the two corpora to annotate some of the *same* sentences in different styles.

But fortunately, we can automatically obtain a noisy form of the necessary paired-tree training data. A parser trained on the source corpus can parse the sentences in our target corpus, yielding trees (or more generally, probability distributions over trees) in the source style. We will then learn a tree transformation model relating these noisy source trees to our known trees in the target style.

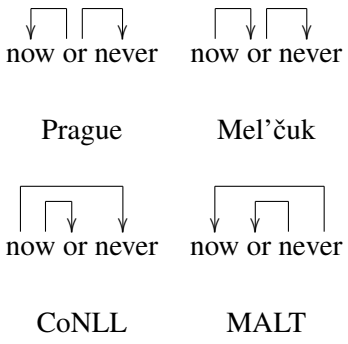

Figure 1: Four of the five logically possible schemes for annotating coordination show up in human-produced dependency treebanks. (The other possibility is a reverse Mel'čuk scheme.) These treebanks also differ on other conventions.

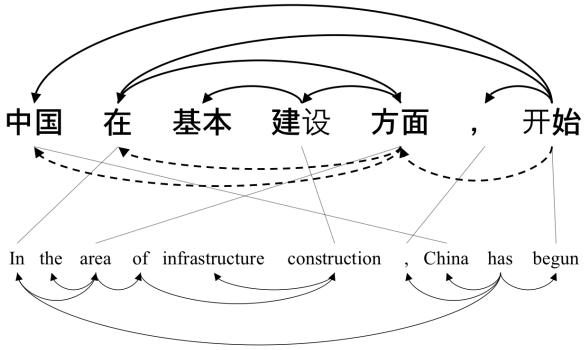

Figure 2: With the English tree and alignment provided by a parser and aligner at test time, the Chinese parser finds the correct dependencies (see §6). A monolingual parser's incorrect edges are shown with dashed lines.

This model should enable us to convert the original large source corpus to target style, giving us additional training data in the target style.

1.2 Parser Projection

For many target languages, however, we do not have the luxury of a large parsed "source corpus" in the language, even one in a different style or domain as above. Thus, we may seek other forms of data to augment our small target corpus. One option would be to leverage unannotated text (McClosky et al., 2006; Smith and Eisner, 2007). But we can also try to transfer syntactic information from a parsed source corpus *in another language*. This is an extreme case of out-of-domain data. This leads to the second task of this paper: *learning a statistical model to transform a syntactic analysis of a sentence in one language into an analysis of its translation*.

Tree transformations are often modeled with *synchronous* grammars. Suppose we are given a sentence w' in the "source" language and its translation w into the "target" language. Their syntactic parses t' and t are presumably not independent, but will tend to have some parallel or at least correlated structure. So we could *jointly* model the parses t', t and the alignment a between them, with a model of the form $p(t, a, t' \mid w, w')$.

Such a joint model captures how t, a, t' mutually constrain each other, so that even partial knowledge of some of these three variables can help us to recover the others when training or decoding on bilingual text. This idea underlies a number of recent papers on syntax-based alignment (using t and t' to better recover a), grammar induction from bitext (using a to better recover t and t'), parser projection (using t' and a to better recover t), as well as full joint parsing (Smith and Smith, 2004; Burkett and Klein, 2008).

In this paper, we condition on the 1-best source tree t'. As for the alignment a, our models either condition on the 1-best alignment or integrate the alignment out. Our models are thus of the form $p(t \mid w, w', t', a)$ or, in the generative case, $p(w, t, a \mid w', t')$. We intend to consider other formulations in future work.

So far, this is very similar to the monolingual parser adaptation scenario, but there are a few key differences. Since the source and target sentences in the bitext are in different languages, there is no longer a trivial alignment between the words of the source and target trees. Given word alignments, we could simply try to project dependency links in the source tree onto the target text. A link-by-link projection, however, could result in invalid trees on the target side, with cycles or disconnected words. Instead, our models learn the necessary transformations that align and transform a source tree into a target tree by means of quasi-synchronous grammar (QG) features.

Figure 2 shows an example of bitext helping disambiguation when a parser is trained with only a small number of Chinese trees. With the help of the English tree and alignment, the parser is able to recover the correct Chinese dependencies using QG features. Incorrect edges from the monolingual parser are shown with dashed lines. (The bilingual parser corrects additional errors in the second half of this sentence, which has been removed to improve legibility.) The parser is able to recover the long-distance dependency from the first Chinese word (*China*) to the last (*begun*), while skipping over the intervening noun

phrase that confused the undertrained monolingual parser. Although, due to the auxiliary verb, "China" and "begun" are *siblings* in English and not in direct dependency, the QG features still leverage this indirect projection.

1.3 Plan of the Paper

We start by describing the features we use to augment conditional and generative parsers when scoring pairs of trees (§2). Then we discuss in turn monolingual (§3) and cross-lingual (§4) parser adaptation. Finally, we present experiments on cross-lingual parser projection in conditions when no target language trees are available for training (§5) and when some trees are available (§6).

2 Form of the Model

What should our model of source and target trees look like? In our view, traditional approaches based on synchronous grammar are problematic both computationally and linguistically. Full inference takes $O(n^6)$ time or worse (depending on the grammar formalism). Yet synchronous models only consider a limited hypothesis space: e.g., parses must be projective, and alignments must decompose according to the recursive parse structure. (For example, two nodes can be aligned only if their respective parents are also aligned.) The synchronous model's probability mass function is also restricted to decompose in this way, so it makes certain conditional independence assumptions; put another way, it can evaluate only certain properties of the triple (t, a, t').

We instead model (t, a, t') as an *arbitrary graph* that includes dependency links among the words of each sentence as well as arbitrary alignment links between the words of the two sentences. This permits non-synchronous and many-to-many alignments. The only hard constraint we impose is that the dependency links within each sentence must constitute a valid monolingual parse—a directed projective spanning tree.[1]

Given the two sentences w, w', our probability distribution over possible graphs considers local features of the parses, the alignment, and both jointly. Thus, we learn what local syntactic configurations tend to occur in each language and how they correspond across languages. As a result, we might learn that parses are "mostly synchronous," but that there are some systematic cross-linguistic divergences and some instances of sloppy (non-parallel or inexact) translation. Our model is thus a form of quasi-synchronous grammar (QG) (Smith and Eisner, 2006a). In that paper, QG was applied to word alignment and has since found applications in question answering (Wang et al., 2007), paraphrase detection (Das and Smith, 2009), and machine translation (Gimpel and Smith, 2009).

All the models in this paper are conditioned on the source tree t'. Conditionally-trained models of adaptation and projection also condition on the target string w and its alignment a to w' and thus have the form $p(t \mid w, w', t', a)$; the unsupervised, generative projection models in §5 have the form $p(w, t, a \mid w', t')$.

The score s of a given tuple of trees, words, and alignment can thus be written as a dot product of weights $\mathbf{w}$ with features $\mathbf{f}$ and $\mathbf{g}$:

$$s(t, t', a, w, w') = \sum_i w_i f_i(t, w) + \sum_j w_j g_j(t, t', a, w, w')$$

The features $\mathbf{f}$ look only at target words and dependencies. In the conditional models of §3 and §6, these features are those of an edge-factored dependency parser (McDonald et al., 2005). In the generative models of §5, $\mathbf{f}$ has the form of a dependency model with valence (Klein and Manning, 2004). All models, for instance, have a feature template that considers the parts of speech of a potential parent-child relation.

In order to benefit from the source language, we also need to include bilingual features $\mathbf{g}$. When scoring a candidate target dependency link from word $x \rightarrow y$, these features consider the relationship of their corresponding source words x' and y'. (The correspondences are determined by the alignment a.) For instance, the source tree t' may contain the link $x' \rightarrow y'$, which would cause a feature for *monotonic* projection to fire for the $x \rightarrow y$ edge. If, on the other hand, $y' \rightarrow x' \in t'$, a *head-swapping* feature fires. If $x' = y'$, i.e. x and y align to the same word, the *same-word* feature fires. Similar features fire when x' and y' are in grandparent-grandchild, sibling, c-command, or none-of-the above relationships, or when y aligns to NULL. These alignment classes are called *configurations* (Smith and Eisner, 2006a, and following). When training is conditioned on the target words (see §3 and §6 below), we conjoin these

[1] Non-projective parsing would also be possible.

configuration features with the part of speech and coarse part of speech of one or both of the source and target words, i.e. the feature template has from one to four tags.

In conditional training, the exponentiated scores s are normalized by a constant: $Z = \sum_t \exp[s(t, t', a, w, w')]$. For the generative model, the locally normalized generative process is explained in §5.3.4.

Previous researchers have written fix-up rules to massage the projected links after the fact and learned a parser from the resulting trees (Hwa et al., 2005). Instead, our models learn the necessary transformations that align and transform a source tree into a target tree. Other researchers have tackled the interesting task of learning parsers from unparsed bitext alone (Kuhn, 2004; Snyder et al., 2009); our methods take advantage of investments in high-resource languages such as English. In work most closely related to this paper, Ganchev et al. (2009) constrain the posterior distribution over target-language dependencies to align to source dependencies some "reasonable" proportion of the time ($\approx 70\%$, cf. Table 2 in this paper). This approach performs well but cannot directly learn regular cross-language non-isomorphisms; for instance, some fixup rules for auxiliary verbs need to be introduced. Finally, Huang et al. (2009) use features, somewhat like QG configurations, on the shift-reduce actions in a monolingual, target-language parser.

3 Adaptation

As discussed in §1, the adaptation scenario is a special case of parser projection where the word alignments are one-to-one and observed. To test our handling of QG features, we performed experiments in which training saw the correct parse trees in both source and target domains, and the mapping between them was simple and regular. We also performed experiments where the source trees were replaced by the noisy output of a trained parser, making the mapping more complex and harder to learn.

We used the subset of the Penn Treebank from the CoNLL 2007 shared task and converted it to dependency representation while varying two parameters: (1) CoNLL vs. Prague coordination style (Figure 1), and (2) preposition the head vs. the child of its nominal object.

We trained an edge-factored dependency parser (McDonald et al., 2005) on "source" domain data that followed one set of dependency conventions. We then trained an edge-factored parser with QG features on a small amount of "target" domain data. The source parser outputs were produced for all target data, both training and test, so that features for the target parser could refer to them.

In this task, we know what the gold-standard source language parses are for any given text, since we can produce them from the original Penn Treebank. We can thus measure the contribution of adaptation loss alone, and the combined loss of imperfect source-domain parsing with adaptation (Table 1). When no target domain trees are available, we simply have the performance of the source domain parser on this out-of-domain data. Training a target-domain parser on as few as 10 sentences shows substantial improvements in accuracy. In the "gold" conditions, where the target parser starts with perfect source trees, accuracy approaches 100%; in the realistic "parse" conditions, where the target-domain parser gets noisy source-domain parses, the improvements are quite significant but approach a lower ceiling imposed by the performance of the source parser.[2]

The adaptation problem in this section is a simple proof of concept of the QG approach; however, more complex and realistic adaptation problems exist. Monolingual adaptation is perhaps most obviously useful when the source parser is a black-box or rule-based system or is trained on unavailable data. One might still want to use such a parser in some new context, which might require new data or a new annotation standard.

We are also interested in scenarios where we want to avoid expensive retraining on large reannotated treebanks. We would like a linguist to be able to annotate a few trees according to a hypothesized theory and then quickly use QG adaptation to get a parser for that theory. One example would be adapting a constituency parser to produce dependency parses. We have concentrated here on adapting between two dependency parse styles, in order to line up with the cross-lingual tasks to which we now turn.

[2] In the diagonal cells, source and target styles match, so training the QG parser amounts to a "stacking" technique (Martins et al., 2008). The small training size and overregularization of the QG parser mildly hurts in-domain parsing performance.

	% Dependency Accuracy on Target											
	CoNLL-PrepHead			CoNLL-PrepChild			Prague-PrepHead			Prague-PrepChild		
Source	0	10	100	0	10	100	0	10	100	0	10	100
Gold CoNLL-PrepHead	100	99.6	99.6	79.5	96.9	97.8	90.5	95.0	98.1	71.0	92.7	95.4
Parse CoNLL-PrepHead	89.5	88.9	89.0	71.4	85.9	87.9	82.5	84.3	87.8	65.2	82.2	86.1
Gold CoNLL-PrepChild	79.5	96.6	97.3	100	99.6	99.6	71.0	91.3	95.5	89.9	94.5	97.9
Parse CoNLL-PrepChild	71.0	84.2	86.8	88.1	87.5	88.0	64.9	80.7	84.9	80.9	83.5	86.1
Gold Prague-PrepHead	90.5	95.5	96.7	71.0	92.0	94.2	100	99.6	99.6	79.6	97.4	98.1
Parse Prague-PrepHead	83.0	87.1	87.4	65.6	84.2	85.9	88.5	88.3	88.0	70.7	86.4	86.8
Gold Prague-PrepChild	71.0	91.6	93.8	89.9	95.6	96.4	79.6	96.0	97.1	100	99.6	99.6
Parse Prague-PrepChild	65.3	81.7	84.6	81.2	84.5	86.1	70.4	83.2	85.3	86.9	86.1	86.8

Table 1: Adapting a parser to a new annotation style. We learn to parse in a "target" style (wide column label) given some number (narrow column label) of supervised target-style training sentences. As a font of additional features, *all training and test sentences* have already been augmented with parses in some "source" style (row label): either gold-standard parses (an oracle experiment) or else the output of a parser trained on 18k source trees (more realistic). If we have 0 training sentences, we simply output the source-style parse. But with 10 or 100 target-style training sentences, each off-diagonal block learns to adapt, mostly closing the gap with the diagonal block in the same column. In the diagonal blocks, source and target styles match, and the QG parser degrades performance when acting as a "stacked" parser.

4 Cross-Lingual Projection: Background

As in the adaptation scenario above, many syntactic structures can be transferred from one language to another. In this section, we evaluate the extent of this direct projection on a small hand-annotated corpus. In §5, we will use a QG generative model to learn dependency parsers from bitext when there are no annotations in the target language. Finally, in §6,we show how QG features can augment a target-language parser trained on a small set of labeled trees.

For syntactic annotation projection to work at all, we must hypothesize, or observe, that at least some syntactic structures are preserved in translation. Hwa et al. (2005) have called this intuition the **Direct Correspondence Assumption** (DCA, with slight notational changes):

> Given a pair of sentences w and w' that are translations of each other with syntactic structure t and t', if nodes x' and y' of t' are aligned with nodes x and y of t, respectively, and if syntactic relationship $R(x', y')$ holds in t', then $R(x, y)$ holds in t.

The validity of this assumption clearly depends on the node-to-node alignment of the two trees. We again work in a dependency framework, where syntactic nodes are simply lexical items. This allows us to use existing work on word alignment.

Hwa et al. (2005) tested the DCA under idealized conditions by obtaining hand-corrected dependency parse trees of a few hundred sentences of Spanish-English and Chinese-English bitext. They also used human-produced word alignments.

Corpus	Prec.[%]	Rec.[%]
Spanish	64.3	28.4
(no punc.)	72.0	30.8
Chinese	65.1	11.1
(no punc.)	68.2	11.5

Table 2: Precision and recall of direct dependency projection via one-to-one links alone.

Since their word alignments could be many-to-many, they gave a heuristic Direct Projection Algorithm (DPA) for resolving them into component dependency relations. It should be noted that this process introduced empty words into the projected target language tree and left words that are unaligned to English detached from the tree; as a result, they measured performance in dependency F-score rather than accuracy. With manual English parses and word alignments, this DPA achieved 36.8% F-score in Spanish and 38.1% in Chinese. With Collins-model English parses and GIZA++ word alignments, F-score was 33.9% for Spanish and 26.3% for Chinese. Compare this to the Spanish attach-left baseline of 31.0% and the Chinese attach-right baselines of 35.9%. These discouragingly low numbers led them to write language-specific transformation rules to fix up the projected trees. After these rules were applied to the projections of automatic English parses, F-score was 65.7% for English and 52.4% for Chinese.

While these F-scores were low, it is useful to look at a subset of the alignment: dependencies projected across one-to-one alignments before the heuristic fix-ups had a much higher precision, if lower recall, than Hwa et al.'s final results. Us-

ing Hwa et al.'s data, we calculated that the precision of projection to Spanish and Chinese via these one-to-one links was $\approx 65\%$ (Table 2). There is clearly more information in these direct links than one would think from the F-scores. To exploit this information, however, we need to overcome the problems of (1) learning from partial trees, when not all target words are attached, and (2) learning in the presence of the still considerable noise in the projected one-to-one dependencies—e.g., at least 28% error for Spanish non-punctuation dependencies.

What does this noise consist of? Some errors reflect fairly arbitrary annotation conventions in treebanks, e.g. should the auxiliary verb govern the main verb or vice versa. (Examples like this suggest that the projection problem contains the adaptation problem above.) Other errors arise from divergences in the complements required of certain head words. In the German-English translation pair, with co-indexed words aligned,

$$[\text{an } [\text{den Libanon}_1]]\ \text{denken}_2 \ \leftrightarrow\ \text{remember}_2\ \text{Lebanon}_1$$

we would prefer that the preposition *an* attach to *denken*, even though the preposition's object *Libanon* aligns to a direct child of *remember*. In other words, we would like the grandparent-parent-child chain of *denken* $\rightarrow$ *an* $\rightarrow$ *Libanon* to align to the parent-child pair of *remember* $\rightarrow$ *Lebanon*. Finally, naturally occurring bitexts contain some number of free or erroneous translations. Machine translation researchers often seek to strike these examples from their training corpora; "free" translations are not usually welcome from an MT system.

5 Unsupervised Cross-Lingual Projection

First, we consider the problem of parser projection when there are zero target-language trees available. As in much other work on unsupervised parsing, we try to learn a generative model that can predict target-language sentences. Our novel contribution is to *condition the probabilities of the generative actions* on the dependency parse of a source-language translation. Thus, our generative model is a quasi-synchronous grammar, exactly as in (Smith and Eisner, 2006a).[3]

When training on target sentences w, therefore, we tune the model parameters to maximize not $\sum_t p(t, w)$ as in ordinary EM, but rather $\sum_t p(t, w, a \mid t', w')$. We hope that this *conditional EM* training will drive the model to posit appropriate syntactic relationships in the latent variable t, because—thanks to the structure of the QG model—that is the easiest way for it to exploit the extra information in t', w' to help predict w.[4] At test time, t', w' are not made available, so we just use the trained model to find $\text{argmax}_t\, p(t \mid w)$, backing off from the conditioning on t', w' and summing over a.

Below, we present the specific generative model (§5.1) and some details of training (§5.2). We will then compare three approaches (§5.3):

§**5.3.2** a straight EM baseline (which does not condition on t', w' at all)

§**5.3.3** a "hard" projection baseline (which naively projects t', w' to derive direct supervision in the target language)

§**5.3.4** our conditional EM approach above (which makes t', w' available to the learner for "soft" indirect supervision via QG)

5.1 Generative Models

Our base models of target-language syntax are generative dependency models that have achieved state-of-the art results in unsupervised dependency structure induction. The simplest version, called Dependency Model with Valence (DMV), has been used in isolation and in combination with other models (Klein and Manning, 2004; Smith and Eisner, 2006b). The DMV generates the right children, and then independently the left children, for each node in the dependency tree. Nodes correspond to words, which are represented by their part-of-speech tags. At each step of generation, the DMV stochastically chooses whether to stop generating, conditioned on the currently generating head; whether it is generating to the right or left; and whether it has yet generated any children on that side. If it chooses to continue, it then

[3] Our task here is new; they used it for alignment.

[4] The contrastive estimation of Smith and Eisner (2005) also used a form of conditional EM, with similar motivation. They suggested that EM grammar induction, which learns to predict w, unfortunately learns mostly to predict lexical topic or other properties of the training sentences that do not strongly require syntactic latent variables. To focus EM on modeling the *syntactic* relationships, they conditioned the prediction of w on almost complete knowledge of the lexical items. Similarly, we condition on a source translation of w. Furthermore, our QG model structure makes it easy for EM to learn to exploit the (explicitly represented) syntactic properties of that translation when predicting w.

stochastically generates the tag of a new child, conditioned on the head. The parameters of the model are thus of the form

$$p(stop \mid head, dir, adj) \quad (1)$$

$$p(child \mid head, dir) \quad (2)$$

where $head$ and $child$ are part-of-speech tags, $dir \in \{\text{left}, \text{right}\}$, and $adj, stop \in \{\text{true}, \text{false}\}$. ROOT is stipulated to generate a single right child.

Bilingual configurations that condition on t', w' (§2) are incorporated into the generative process as in Smith and Eisner (2006a). When the model is generating a new child for word x, aligned to x', it first chooses a configuration and then chooses a source word y' in that configuration. The child y is then generated, conditioned on its parent x, most recent sibling a, and its source analogue y'.

5.2 Details of EM Training

As in previous work on grammar induction, we learn the DMV from part-of-speech-tagged target-language text. We use expectation maximization (EM) to maximize the likelihood of the data. Since the likelihood function is nonconvex in the unsupervised case, our choice of initial parameters can have a significant effect on the outcome. Although we could also try many random starting points, the initializer in Klein and Manning (2004) performs quite well.

The base dependency parser generates the right dependents of a head separately from the left dependents, which allows $O(n^3)$ dynamic programming for an n-word target sentence. Since the QG annotates nonterminals of the grammar with single nodes of t', and we consider two nodes of t' when evaluating the above dependency configurations, QG parsing runs in $O(n^3m^2)$ for an m-word source sentence. If, however, we restrict candidate senses for a target child c to come from links in an IBM Model 4 Viterbi alignment, we achieve $O(n^3k^2)$, where k is the maximum number of possible words aligned to a given target language word. In practice, $k \ll m$, and parsing is not appreciably slower than in the monolingual setting.

If all configurations were equiprobable, the source sentence would provide no information to the target. In our QG experiments, therefore, we started with a bias towards direct parent–child links and a very small probability for breakages of locality. The values of other configuration parameters seem, experimentally, less important for insuring accurate learning.

5.3 Experiments

Our experiments compare learning on target language text to learning on parallel text. In the latter case, we compare learning from high-precision one-to-one alignments alone, to learning from all alignments using a QG.

5.3.1 Corpora

Our development and test data were drawn from the German TIGER and Spanish Cast3LB treebanks as converted to projective dependencies for the CoNLL 2007 Shared Task (Brants et al., 2002; Civit Torruella and Martí Antonín, 2002).[5]

Our training data were subsets of the 2006 Statistical Machine Translation Workshop Shared Task, in particular from the German-English and Spanish-English Europarl parallel corpora (Koehn, 2002). The Shared Task provided pre-built automatic GIZA++ word alignments, which we used to facilitate replicability. Since these word alignments do not contain posterior probabilities or null links, nor do they distinguish which links are in the IBM Model intersection, we treated all links as equally likely when learning the QG. Target language words unaligned to any source language words were the only nodes allowed to align to NULL in QG derivations.

We parsed the English side of the bitext with the projective dependency parser described by McDonald et al. (2005) trained on the Penn Treebank §§2–20. Much previous work on unsupervised grammar induction has used gold-standard part-of-speech tags (Smith and Eisner, 2006b; Klein and Manning, 2004; Klein and Manning, 2002). While there are no gold-standard tags for the Europarl bitext, we did train a conditional Markov

[5] We made one change to the annotation conventions in German: in the dependencies provided, words in a noun phrase governed by a preposition were all attached to that preposition. This meant that in the phrase *das Kind* ("the child") in, say, subject position, *das* was the child of *Kind*; but, in *für das Kind* ("for the child"), *das* was the child of *für*. This seems to be a strange choice in converting from the TIGER constituency format, which does in fact annotate NPs inside PPs; we have standardized prepositions to govern only the head of the noun phrase. We did *not* change any other annotation conventions to make them more like English. In the Spanish treebank, for instance, control verbs are the children of their verbal complements: in *quiero decir* ("I want to say"="I mean"), *quiero* is the child of *decir*. In German coordinations, the coordinands all attach to the first, but in English, they all attach to the last. These particular divergences in annotation style hurt all of our models equally (since none of them have access to labeled trees). These annotation divergences are one motivation for experiments below that include some target trees.

Baselines	Dependency accuracy [%] German	Spanish
Modify prev.	18.2	28.5
Modify next	27.5	21.4
EM	30.2	25.6
Hard proj.	66.2	59.1
Hard proj. w/EM	58.6	53.0
QG w/EM	**68.5**	**64.8**

Table 3: Test accuracy with unsupervised training methods

model tagger on a few thousand tagged sentences. This is the only supervised data we used in the target. We created versions of each training corpus with the first thousand, ten thousand, and hundred thousand sentence pairs, each a prefix of the next. Since the target-language-only baseline converged much more slowly, we used a version of the corpora with sentences 15 target words or fewer.

5.3.2 Fully Unsupervised EM

Using the target side of the bitext as training data, we initialized our model parameters as described in §5.2 and ran EM. We checked convergence on a development set and measured unlabeled dependency accuracy on held-out test data. We compare performance to simple attach-right and attach left baselines (Table 3). For mostly head-final German, the "modify next" baseline is better; for mostly head-initial Spanish, "modify previous" wins. Even after several hundred iterations, performance was slightly, but not significantly better than the baseline for German. EM training did not beat the baseline for Spanish.[6]

5.3.3 Hard Projection, Semi-Supervised EM

The simplest approach to using the high-precision one-to-one word alignments is labeled "hard projection" in the table. We filtered the training corpus to find sentences where enough links were projected to completely determine a target language tree. Of course, we needed to filter more than 1000 sentences of bitext to output 1000 training sentences in this way. We simply perform supervised training with this subset, which is still quite noisy (§4), and performance quickly plateaus. Still, this method substantially improves over the baselines and unsupervised EM.

Restricting ourselves to fully projected trees seems a waste of information. We can also simply take all one-to-one projected links, impute expected counts for the remaining dependencies with EM, and update our models. This approach ("hard projection with EM"), however, performed worse than using only the fully projected trees. In fact, only the first iteration of EM with this method made any improvement; afterwards, EM degraded accuracy further from the numbers in Table 3.

5.3.4 Soft Projection: QG & Conditional EM

The quasi-synchronous model used all of the alignments in re-estimating its parameters and performed significantly better than hard projection. Unlike EM on the target language alone, the QG's performance does not depend on a clever initializer for initial model weights—all parameters of the generative model except for the QG configuration features were initialized to zero. Setting the prior to prefer direct correspondence provides the necessary bias to initialize learning.

Error analysis showed that certain types of dependencies eluded the QG's ability to learn from bitext. The Spanish treebank treats some verbal complements as the heads of main verbs and auxiliary verbs as the children of participles; the QG, following the English, learned the opposite dependency direction. Spanish treebank conventions for punctuation were also a common source of errors. In both German and Spanish, coordinations (a common bugbear for dependency grammars) were often mishandled: both treebanks attach the later coordinands and any conjunctions to the first coordinand; the reverse is true in English. Finally, in both German and Spanish, preposition attachments often led to errors, which is not surprising given the unlexicalized target-language grammars. Rather than trying to adjudicate which dependencies are "mere" annotation conventions, it would be useful to test learned dependency models on some extrinsic task such as relation extraction or machine translation.

6 Supervised Cross-Lingual Projection

Finally, we consider the problem of parser projection when some target language trees are available. As in the adaptation case (§3), we train a conditional model (*not* a generative DMV) of the target

[6] While these results are worse than those obtained previously for this model, the experiments in Klein and Manning (2004) and only used sentences of 10 words or fewer, without punctuation, and with gold-standard tags. Punctuation in particular seems to trip up the initializer: since a sentence-final periods appear in most sentences, EM often decides to make it the head.

tree given the target sentence, using the monolingual and bilingual QG features, including configurations conjoined with tags, outlined above (§2).

For these experiments, we used the LDC's English-Chinese Parallel Treebank (ECTB). Since manual word alignments also exist for a part of this corpus, we were able to measure the loss in accuracy (if any) from the use of an automatic English parser and word aligner. The source-language English dependency parser was trained on the Wall Street Journal, where it achieved 91% dependency accuracy on development data. However, it was only 80.3% accurate when applied to our task, the English side of the ECTB.[7]

After parsing the source side of the bitext, we train a parser on the annotated target side, using QG features described above (§2). Both the monolingual target-language parser and the projected parsers are trained to optimize conditional likelihood of the target trees t' with ten iterations of stochastic gradient ascent.

In Figure 3, we plot the performance of the target-language parser on held-out bitext. Although projection performance is, not surprisingly, better if we know the true source trees at training and test time, even with the 1-best output of the source parser, QG features help produce a parser as accurate asq one trained on twice the amount of monolingual data. In ablation experiments, we included bilingual features only for directly projected links, with no features for head-swapping, grandparents, etc. When using 1-best English parses, parsers trained only with direct-projection and monolingual features performed worse; when using gold English parses, parsers with direct-projection-only features performed better when trained with more Chinese trees.

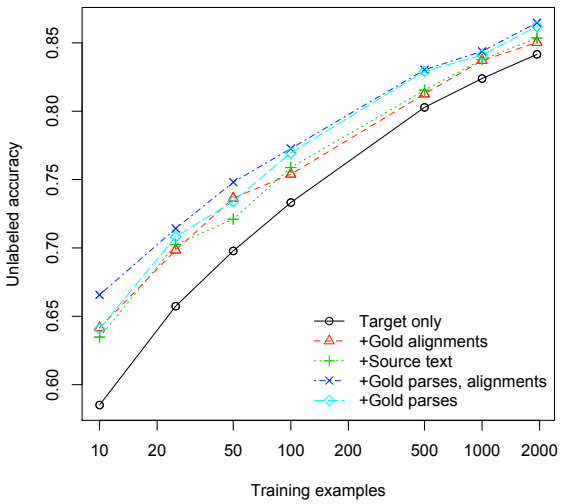

Figure 3: Parser projection with target trees. Using the true or 1-best parse trees in the source language is equivalent to having twice as much data in the target language. Note that the penalty for using automatic alignments instead of gold alignments is negligible; in fact, using *Source text* alone is often higher than +*Gold alignments*. Using gold source trees, however, significantly outperforms using 1-best source trees.

7 Discussion

The two related problems of parser adaptation and projection are often approached in different ways. Many adaptation methods operate by simple augmentations of the target feature space, as we have done here (Daume III, 2007). Parser projection, on the other hand, often uses a multi-stage pipeline (Hwa et al., 2005). The methods presented here move parser projection much closer in efficiency and simplicity to monolingual parsing.

We showed that augmenting a target parser with quasi-synchronous features can lead to significant improvements—first in experiments with adapting to different dependency representations in English, and then in cross-language parser projection. As with many domain adaptation problems, it is quite helpful to have some annotated target data, especially when annotation styles vary (Dredze et al., 2007). Our experiments show that unsupervised QG projection improves on parsers trained using only high-precision projected annotations and far outperforms, by more than 35% absolute dependency accuracy, unsupervised EM. When a small number of target-language parse trees is available, projection gives a boost equivalent to doubling the number of target trees.

The loss in performance from conditioning only on noisy 1-best source parses points to some natural avenues for improvement. We are exploring methods that incorporate a packed parse forest on the source side and similar representations of uncertainty about alignments. Building on our recent belief propagation work (Smith and Eisner, 2008), we can jointly infer two dependency trees and their alignment, under a joint distribution $p(t, a, t' \mid w, w')$ that evaluates the full graph of dependency and alignment edges.

[7] It would be useful to explore whether the techniques of §3 above could be used to improve English accuracy by domain adaptation. Furthermore, a model with QG features trained to perform well on Chinese should not suffer from an inaccurate, but consistent, English parser, but the results in Figure 3 indicate a significant benefit to be had from better English parsing or from joint Chinese-English inference.

References

S. Brants, S. Dipper, S. Hansen, W. Lezius, and G. Smith. 2002. The TIGER treebank. In *TLT*.

David Burkett and Dan Klein. 2008. Two languages are better than one (for syntactic parsing). In *EMNLP*.

M. Civit Torruella and M. A. Martí Antonín. 2002. Design principles for a Spanish treebank. In *TLT*.

Dipanjan Das and Noah A. Smith. 2009. Paraphrase identification as probabilistic quasi-synchronous recognition. In *ACL-IJCNLP*.

Hal Daume III. 2007. Frustratingly easy domain adaptation. In *ACL*, pages 256–263.

Mark Dredze, John Blitzer, Partha Pratim Talukdar, Kuzman Ganchev, João Graca, and Fernando Pereira. 2007. Frustratingly hard domain adaptation for dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 1051–1055.

Kuzman Ganchev, Jennifer Gillenwater, and Ben Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *ACL-IJCNLP*.

Kevin Gimpel and Noah A. Smith. 2009. Feature-rich translation by quasi-synchronous lattice parsing. In *EMNLP*.

Liang Huang, Wenbin Jiang, and Qun Liu. 2009. Bilingually-constrained (monolingual) shift-reduce parsing. In *EMNLP*.

Rebecca Hwa, Philip Resnik, Amy Weinberg, Clara Cabezas, and Okan Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11:311–325.

Dan Klein and Christopher D. Manning. 2002. A generative constituent-context model for improved grammar induction. In *ACL*, pages 128–135.

Dan Klein and Christopher D. Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *ACL*, pages 479–486.

Philipp Koehn. 2002. Europarl: A multilingual corpus for evaluation of machine translation. http://www.iccs.informatics.ed.ac.uk/~pkoehn/-publications/europarl.ps.

Jonas Kuhn. 2004. Experiments in parallel-text based grammar induction. In *ACL*, pages 470–477.

André F. T. Martins, Dipanjan Das, Noah A. Smith, and Eric P. Xing. 2008. Stacking dependency parsers. In *EMNLP*.

David McClosky, Eugene Charniak, and Mark Johnson. 2006. Reranking and self-training for parser adaptation. In *ACL*, pages 337–344.

Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005. Online large-margin training of dependency parsers. In *ACL*, pages 91–98.

Noah A. Smith and Jason Eisner. 2005. Guiding unsupervised grammar induction using contrastive estimation. In *International Joint Conference on Artificial Intelligence (IJCAI) Workshop on Grammatical Inference Applications*, Edinburgh, July.

David A. Smith and Jason Eisner. 2006a. Quasi-synchronous grammars: Alignment by soft projection of syntactic dependencies. In *Proceedings of the HLT-NAACL Workshop on Statistical Machine Translation*, pages 23–30.

Noah A. Smith and Jason Eisner. 2006b. Annealing structural bias in multilingual weighted grammar induction. In *ACL-COLING*, pages 569–576.

David A. Smith and Jason Eisner. 2007. Bootstrapping feature-rich dependency parsers with entropic priors. In *EMNLP-CoNLL*, pages 667–677.

David A. Smith and Jason Eisner. 2008. Dependency parsing by belief propagation. In *EMNLP*, pages 145–156.

David A. Smith and Noah A. Smith. 2004. Bilingual parsing with factored estimation: Using English to parse Korean. In *EMNLP*, pages 49–56.

Benjamin Snyder, Tahira Naseem, and Regina Barzilay. 2009. Unsupervised multilingual grammar induction. In *ACL-IJCNLP*.

Mengqiu Wang, Noah A. Smith, and Teruko Mitamura. 2007. What is the Jeopardy model? a quasi-synchronous grammar for QA. In *EMNLP-CoNLL*, pages 22–32.