Multi-Source Transfer of Delexicalized Dependency Parsers

Ryan McDonald
Google
New York, NY
ryanmcd@google.com

Slav Petrov
Google
New York, NY
slav@google.com

Keith Hall
Google
Zürich
kbhall@google.com

Abstract

We present a simple method for transferring dependency parsers from source languages with labeled training data to target languages without labeled training data. We first demonstrate that delexicalized parsers can be directly transferred between languages, producing significantly higher accuracies than unsupervised parsers. We then use a constraint driven learning algorithm where constraints are drawn from parallel corpora to project the final parser. Unlike previous work on projecting syntactic resources, we show that simple methods for introducing multiple source languages can significantly improve the overall quality of the resulting parsers. The projected parsers from our system result in state-of-the-art performance when compared to previously studied unsupervised and projected parsing systems across eight different languages.

1 Introduction

Statistical parsing has been one of the most active areas of research in the computational linguistics community since the construction of the Penn Treebank (Marcus et al., 1993). This includes work on phrase-structure parsing (Collins, 1997; Charniak, 2000; Petrov et al., 2006), dependency parsing (McDonald et al., 2005; Nivre et al., 2006) as well as a number of other formalisms (Clark and Curran, 2004; Wang and Harper, 2004; Shen and Joshi, 2008). As underlying modeling techniques have improved, these parsers have begun to converge to high levels of accuracy for English newswire text. Subsequently, researchers have begun to look at both porting these parsers to new domains (Gildea, 2001; McClosky et al., 2006; Petrov et al., 2010) and constructing parsers for new languages (Collins et al., 1999; Buchholz and Marsi, 2006; Nivre et al., 2007).

One major obstacle in building statistical parsers for new languages is that they often lack the manually annotated resources available for English. This observation has led to a vast amount of research on unsupervised grammar induction (Carroll and Charniak, 1992; Klein and Manning, 2004; Smith and Eisner, 2005; Cohen and Smith, 2009; Berg-Kirkpatrick and Klein, 2010; Naseem et al., 2010; Spitkovsky et al., 2010; Blunsom and Cohn, 2010). Grammar induction systems have seen large advances in quality, but parsing accuracies still significantly lag behind those of supervised systems. Furthermore, they are often trained and evaluated under idealized conditions, e.g., only on short sentences or assuming the existence of gold-standard part-of-speech (POS) tags.[1] The reason for these assumptions is clear. Unsupervised grammar induction is difficult given the complexity of the analysis space. These assumptions help to give the model traction.

The study of unsupervised grammar induction has many merits. Most notably, it increases our understanding of how computers (and possibly humans) learn in the absence of any explicit feedback. However, the gold POS tag assumption weakens any conclusions that can be drawn, as part-of-speech are also a form of syntactic analysis, only shallower. Furthermore, from a practical standpoint, it is rarely the case that we are completely devoid of resources for most languages. This point has been made by

[1] A notable exception is the work of Seginer (2007).

studies that transfer parsers to new languages by projecting syntax across word alignments extracted from parallel corpora (Hwa et al., 2005; Ganchev et al., 2009; Smith and Eisner, 2009). Although again, most of these studies also assume the existence of POS tags.

In this work we present a method for creating dependency parsers for languages for which no labeled training data is available. First, we train a source side English parser that, crucially, is delexicalized so that its predictions rely soley on the part-of-speech tags of the input sentence, in the same vein as Zeman and Resnik (2008). We empirically show that directly transferring delexicalized models (i.e. parsing a foreign language POS sequence with an English parser) already outperforms state-of-the-art unsupervised parsers by a significant margin. This result holds in the presence of both gold POS tags as well as automatic tags projected from English. This emphasizes that even for languages with no syntactic resources – or possibly even parallel data – simple transfer methods can already be more powerful than grammar induction systems.

Next, we use this delexicalized English parser to seed a perceptron learner for the target language. The model is trained to update towards parses that are in high agreement with a source side English parse based on constraints drawn from alignments in the parallel data. We use the augmented-loss learning procedure (Hall et al., 2011) which is closely related to constraint driven learning (Chang et al., 2007; Chang et al., 2010). The resulting parser consistently improves on the directly transferred delexicalized parser, reducing relative errors by 8% on average, and as much as 18% on some languages. Finally, we show that by transferring parsers from multiple source languages we can further reduce errors by 16% over the directly transferred English baseline. This is consistent with previous work on multilingual part-of-speech (Snyder et al., 2009) and grammar (Berg-Kirkpatrick and Klein, 2010; Cohen and Smith, 2009) induction, that shows that adding languages leads to improvements.

We present a comprehensive set of experiments on eight Indo-European languages for which a significant amount of parallel data exists. We make no language specific enhancements in our experiments. We report results for sentences of *all* lengths, as well as with gold and automatically induced part-of-speech tags. We also report results on sentences of length 10 or less with gold part-of-speech tags to compare with previous work. Our results consistently outperform the previous state-of-the-art across all languages and training configurations.

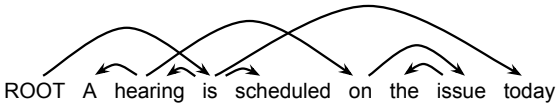

Figure 1: An example (unlabeled) dependency tree.

2 Preliminaries

In this paper we focus on transferring dependency parsers between languages. A dependency parser takes a tokenized input sentence (optionally part-of-speech tagged) and produces a connected tree where directed arcs represent a syntactic head-modifier relationship. An example of such a tree is given in Figure 1. Dependency tree arcs are often labeled with the role of the syntactic relationship, e.g., *is* to *hearing* might be labeled as SUBJECT. However, we focus on unlabeled parsing in order to reduce problems that arise due to different treebank annotation schemes. Of course, even for unlabeled dependencies, significant variations in the annotation schemes remain. For example, in the Danish treebank determiners govern adjectives and nouns in noun phrases, while in most other treebanks the noun is the head of the noun phrase. Unlike previous work (Zeman and Resnik, 2008; Smith and Eisner, 2009), we do not apply any transformations to the treebanks, which makes our results easier to reproduce, but systematically underestimates accuracy.

2.1 Data Sets

The treebank data in our experiments are from the CoNLL shared-tasks on dependency parsing (Buchholz and Marsi, 2006; Nivre et al., 2007). We use English (en) only as a source language throughout the paper. Additionally, we use the following eight languages as both source and target languages: Danish (da), Dutch (nl), German (de), Greek (el), Italian (it), Portuguese (pt), Spanish (es) and Swedish (sv). For languages that were included in both the 2006 and 2007 tasks, we used the treebank from the lat-

ter. We focused on this subset of languages because they are Indo-European and a significant amount of parallel data exists for each language. By presenting results on eight languages our study is already more comprehensive than most previous work in this area. However, the restriction to Indo-European languages does make the results less conclusive when one wishes to transfer a parser from English to Chinese, for example. To account for this, we report additional results in the discussion for non-Indo-European languages. For all data sets we used the predefined training and testing splits.

Our approach relies on a consistent set of part-of-speech tags across languages and treebanks. For this we used the universal tagset from Petrov et al. (2011), which includes: NOUN (nouns), VERB (verbs), ADJ (adjectives), ADV (adverbs), PRON (pronouns), DET (determiners), ADP (prepositions or postpositions), NUM (numerals), CONJ (conjunctions), PRT (particles), PUNC (punctuation marks) and X (a catch-all tag). Similar tagsets are used by other studies on grammar induction and projection (Naseem et al., 2010; Zeman and Resnik, 2008). For all our experiments we replaced the language specific part-of-speech tags in the treebanks with these universal tags.

Like all treebank projection studies we require a corpus of parallel text for each pair of languages we study. For this we used the Europarl corpus version 5 (Koehn, 2005). The corpus was preprocessed in standard ways and word aligned by running six iterations of IBM Model 1 (Brown et al., 1993), followed by six iterations of the HMM model (Vogel et al., 1996) in both directions. We then intersect word alignments to generate one-to-one alignments.

2.2 Parsing Model

All of our parsing models are based on the transition-based dependency parsing paradigm (Nivre, 2008). Specifically, all models use an arc-eager transition strategy and are trained using the averaged perceptron algorithm as in Zhang and Clark (2008) with a beam size of 8. The features used by all models are: the part-of-speech tags of the first four words on the buffer and of the top two words on the stack; the word identities of the first two words on the buffer and of the top word on the stack; the word identity of the syntactic head of the top word on the stack (if available). All feature conjunctions are included. For treebanks with non-projective trees we use the pseudo-projective parsing technique to transform the treebank into projective structures (Nivre and Nilsson, 2005). We focus on using this parsing system for two reasons. First, the parser is near state-of-the-art on English parsing benchmarks and second, and more importantly, the parser is extremely fast to train and run, making it easy to run a large number of experiments. Preliminary experiments using a different dependency parser – MSTParser (McDonald et al., 2005) – resulted in similar empirical observations.

2.3 Evaluation

All systems are evaluated using unlabeled attachment score (UAS), which is the percentage of words (ignoring punctuation tokens) in a corpus that modify the correct head (Buchholz and Marsi, 2006). Furthermore, we evaluate with both gold-standard part-of-speech tags, as well as predicted part-of-speech tags from the projected part-of-speech tagger of Das and Petrov (2011).[2] This tagger relies only on labeled training data for English, and achieves accuracies around 85% on the languages that we consider. We evaluate in the former setting to compare to previous studies that make this assumption. We evaluate in the latter setting to measure performance in a more realistic scenario – when no target language resources are available.

3 Transferring from English

To simplify discussion, we first focus on the most common instantiation of parser transfer in the literature: transferring from English to other languages. In the next section we expand our system to allow for the inclusion of multiple source languages.

3.1 Direct Transfer

We start with the observation that discriminatively trained dependency parsers rely heavily on part-of-speech tagging features. For example, when training and testing a parser on our English data, a parser with all features obtains an UAS of 89.3%[3] whereas

[2] Available at http://code.google.com/p/pos-projection/

[3] The best system at CoNLL 2007 achieved 90.1% and used a richer part-of-speech tagset (Nivre et al., 2007).

a *delexicalized* parser – a parser that only has non-lexical features – obtains an UAS of 82.5%. The key observation is that part-of-speech tags contain a significant amount of information for unlabeled dependency parsing.

This observation combined with our universal part-of-speech tagset, leads to the idea of *direct transfer*, i.e., directly parsing the target language with the source language parser without relying on parallel corpora. This idea has been previously explored by Zeman and Resnik (2008) and recently by Søgaard (2011). Because we use a mapping of the treebank specific part-of-speech tags to a common tagset, the performance of a such a system is easy to measure – simply parse the target language data set with a delexicalized parser trained on the source language data. We conducted two experiments. In the first, we assumed that the test set for each target language had gold part-of-speech tags, and in the second we used predicted part-of-speech tags from the projection tagger of Das and Petrov (2011), which also uses English as the source language.

UAS for all sentence lengths without punctuation are given in Table 1. We report results for both the English direct transfer parser (en-dir.) as well as a baseline unsupervised grammar induction system – the dependency model with valence (DMV) of Klein and Manning (2004), as obtained by the implementation of Ganchev et al. (2010). We trained on sentences of length 10 or less and evaluated on all sentences from the test set.[4] For DMV, we reversed the direction of all dependencies if this led to higher performance. From this table we can see that direct transfer is a very strong baseline and is over 20% absolute better than the DMV model for both gold and predicted POS tags. Table 4, which we will discuss in more detail later, further shows that the direct transfer parser also significantly outperforms state-of-the-art unsupervised grammar induction models, but in a more limited setting of sentences of length less than 10.

Direct transfer works for a couple of reasons. First, part-of-speech tags contain a significant amount of information for parsing unlabeled dependencies. Second, this information can be transferred, to some degree, across languages and treebank standards. This is because, at least for Indo-European languages, there is some regularity in how syntax is expressed, e.g., primarily SVO, prepositional, etc. Even though there are some differences with respect to relative location of certain word classes, strong head-modifier POS tag preferences can still help resolve these, especially when no other viable alternatives are available. Consider for example an artificial sentence with a tag sequence: 'VERB NOUN ADJ DET PUNC'. The English parser still predicts that the NOUN and PUNC modify the VERB and the ADJ and DET modify the NOUN, even though in the English data such noun phrases are unlikely.[5]

3.2 Projected Transfer

Unlike most language transfer systems for parsers, the direct transfer approach does not rely on projecting syntax across aligned parallel corpora (modulo the fact that non-gold tags come from a system that uses parallel corpora). In this section we describe a simple mechanism for projecting from the direct transfer system using large amounts of parallel data in a similar vein to Hwa et al. (2005), Ganchev et al. (2009), Smith and Eisner (2009) inter alia. The algorithm is based on the work of Hall et al. (2011) for training extrinsic parser objective functions and borrows heavily from ideas in learning with weak supervision including work on learning with constraints (Chang et al., 2007) and posterior regularization (Ganchev et al., 2010). In our case, the weak signals come from aligned source and target sentences, and the agreement in their corresponding parses, which is similar to posterior regularization or the bilingual view of Smith and Smith (2004) and Burkett et al. (2010).

The algorithm is given in Figure 2. It starts by labeling a set of target language sentences with a parser, which in our case is the direct transfer parser from the previous section (line 1). Next, it uses these parsed target sentences to 'seed' a new parser by training a parameter vector using the predicted parses as a gold standard via standard perceptron updates for J rounds (lines 3-6). This generates a parser that emulates the direct transfer parser, but

[4] Training on all sentences results in slightly lower accuracies on average.

[5] This requires a transition-based parser with a beam greater than 1 to allow for ambiguity to be resolved at later stages.

Notation:

x:	input sentence
y:	dependency tree
a:	alignment
w:	parameter vector
$\phi(x, y)$:	feature vector
DP:	dependency parser, i.e., $DP : x \rightarrow y$

Input:

$\mathcal{X} = \{x_i\}_{i=1}^{n}$:	target language sentences
$\mathcal{P} = \{(x_i^s, x_i^t, a_i)\}_{i=1}^{m}$:	aligned source-target sentences
DP_{delex}:	delexicalized source parser
DP_{lex}:	lexicalized source parser

Algorithm:

1. Let $\mathcal{X}' = \{(x_i, y_i)\}_{i=1}^{n}$ where $y_i = DP_{\text{delex}}(x_i)$
2. $w = 0$

seed-stage

3. for $j : 1 \ldots J$
4. for $x_i : x_1 \ldots x_n$
5. Let $y = \text{argmax}_y \, w \cdot \phi(x_i, y)$
6. $w = w + \phi(x_t, y_i) - \phi(x_i, y)$

projection-stage

7. for $(x_i^s, x_i^t, a_i) : (x_1^s, x_1^t, a_1) \ldots (x_m^s, x_m^s, a_m)$
8. Let $y_s = DP_{\text{lex}}(x_i^s)$
9. Let $\mathcal{Y}_t = \{y_i^1, \ldots, y_i^k\}$, where:
   $y_i^k = \text{argmax}_{y \notin \{y_i^1, \ldots, y_i^{k-1}\}} \, w \cdot \phi(x_i^t, y)$
10. Let $y_t = \text{argmax}_{y_t \in \mathcal{Y}_t} \text{ALIGN}(y_s, y_t, a_i)$
11. $w = w + \phi(x_i, y_t) - \phi(x_i, y_i^1)$

return DP^* such that $DP^*(x) = \text{argmax}_y \, w \cdot \phi(x, y)$

Figure 2: Perceptron-based learning algorithm for training a parser by seeding the model with a direct transfer parser and projecting constraints across parallel corpora.

has now been lexicalized and is working in the space of target language sentences. Next, the algorithm iterates over the sentences in the parallel corpus. It parses the English sentence with an English parser (line 8, again a lexicalized parser). It then uses the current target language parameter vector to create a k-best parse list for the target sentence (line 9). From this list, it selects the parse whose dependencies align most closely with the English parse via the pre-specified alignment (line 10, also see below for the definition of the ALIGN function). It then uses this selected parse as a proxy to the gold standard parse to update the parameters (line 11).

The intuition is simple. The parser starts with non-random accuracies by emulating the direct transfer model and slowly tries to induce better parameters by selecting parses from its k-best list that are considered 'good' by some external metric. The algorithm then updates towards that output. In this case 'goodness' is determined through the pre-specified sentence alignment and how well the target language parse aligns with the English parse. As a result, the model will, ideally, converge to a state where it predicts target parses that align as closely as possible with the corresponding English parses. However, since we seed the learner with the direct transfer parser, we bias the parameters to select parses that both align well and also have high scores under the direct transfer model. This helps to not only constrain the search space at the start of learning, but also helps to bias dependencies between words that are not part of the alignment.

So far we have not defined the ALIGN function that is used to score potential parses. Let $a = \{(s_{(1)}, t_{(1)}), \ldots, (s_{(n)}, t_{(n)})\}$ be an alignment where $s_{(i)}$ is a word in the source sentence x_s (not necessarily the i^{th} word) and $t_{(i)}$ is similarly a word in the target sentence x_t (again, not necessarily the i^{th} word). The notation $(s_{(i)}, t_{(i)}) \in a$ indicates two words are the i^{th} aligned pair in a. We define the ALIGN function to encode the Direct Correspondence Assumption (DCA) from Hwa et al. (2005):

$$\text{ALIGN}(y_s, y_t, a) = \sum_{\substack{(s_{(i)}, t_{(i)}) \in a \\ (s_{(j)}, t_{(j)}) \in a}} \text{SCORE}(y_s, y_t, (s_{(i)}, s_{(j)}), (t_{(i)}, t_{(j)}))$$

$$\text{SCORE}(y_s, y_t, (s_{(i)}, s_{(j)}), (t_{(i)}, t_{(j)})) = \begin{cases} +1 & \text{if } (s_{(i)}, s_{(j)}) \in y_s \text{ and } (t_{(i)}, t_{(j)}) \in y_t \\ -1 & \text{if } (s_{(i)}, s_{(j)}) \in y_s \text{ and } (t_{(i)}, t_{(j)}) \notin y_t \\ -1 & \text{if } (s_{(i)}, s_{(j)}) \notin y_s \text{ and } (t_{(i)}, t_{(j)}) \in y_t \\ 0 & \text{otherwise} \end{cases}$$

The notation $(i, j) \in y$ indicates that a dependency from head i to modifier j is in tree y. The ALIGN function rewards aligned head-modifier pairs and penalizes unaligned pairs when a possible alignment exists. For all other cases it is agnostic, i.e., when one or both of the modifier or head are not aligned.

Figure 3 shows an example of aligned English-Greek sentences, the English parse and a potential Greek parse. In this case the ALIGN function returns a value of 2. This is because there are three aligned dependencies: *took*→*book*, *book*→*the* and

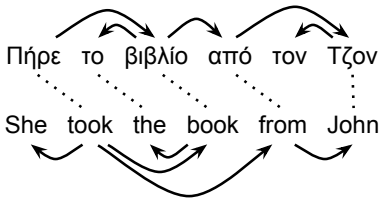

Figure 3: A Greek and English sentence pair. Word alignments are shown as dashed lines, dependency arcs as solid lines.

from→John. These add 3 to the score. There is one incorrectly aligned dependency: the preposition mistakenly modifies the noun on the Greek side. This subtracts 1. Finally, there are two dependencies that do not align: the subject on the English side and a determiner to a proper noun on the Greek side. These do not effect the result.

The learning algorithm in Figure 2 is an instance of augmented-loss training (Hall et al., 2011) which is closely related to the constraint driven learning algorithms of Chang et al. (2007). In that work, external constraints on output structures are used to help guide the learner to good parameter regions. In our model, we use constraints drawn from parallel data exactly in the same manner. Since posterior regularization is closely related to constraint driven learning, this makes our algorithm also similar to the parser projection approach of Ganchev et al. (2009). There are a couple of differences. First, we bias our model towards the direct transfer model, which is already quite powerful. Second, our alignment constraints are used to select parses from a k-best list, whereas in posterior regularization they are used as soft constraints on full model expectations during training. The latter is beneficial as the use of k-best lists does not limit the class of parsers to those whose parameters and search space decompose neatly with the DCA loss function. An empirical comparison to Ganchev et al. (2009) is given in Section 5.

Results are given in Table 1 under the column *en-proj*. For all experiments we train the seed-stage perceptron for 5 iterations ($J = 5$) and we use one hundred times as much parallel data as seed stage non-parallel data ($m = 100n$). The seed-stage non-parallel data is the training portion of each treebank, stripped of all dependency annotations. After training the projected parser we average the parameters

	gold-POS			pred-POS		
	DMV	en-dir.	en-proj.	DMV	en-dir.	en-proj.
da	33.4	45.9	48.2	18.4	44.0	45.5
de	18.0	47.2	50.9	30.3	44.7	47.4
el	39.9	63.9	66.8	21.2	63.0	65.2
es	28.5	53.3	55.8	19.9	50.2	52.4
it	43.1	57.7	60.8	37.7	53.7	56.3
nl	38.5	60.8	67.8	19.9	62.1	66.5
pt	20.1	69.2	71.3	21.0	66.2	67.7
sv	44.0	58.3	61.3	33.8	56.5	59.7
avg	33.2	57.0	60.4	25.3	55.0	57.6

Table 1: UAS for the unsupervised DMV model (DMV), a delexicalized English direct transfer parser (en-dir.) and a English projected parser (en-proj.). Measured on all sentence lengths for both gold and predicted part-of-speech tags as input.

of the model (Collins, 2002). The parsers evaluated using predicted part-of-speech tags use the predicted tags at both training and testing time and are thus free of any target language specific resources.

When compared with the direct transfer model (en-dir. in Table 1), we can see that there is an improvement for every single language, reducing relative error by 8% on average (57.0% to 60.4%) and up to 18% for Dutch (60.8 to 67.8%). One could wonder whether the true power of the projection model comes from the re-lexicalization step – lines 3-6 of the algorithm. However, if just this step is run, then the average UAS only increases from 57.0% to 57.4%, showing that most of the improvement comes from the projection stage. Note that the results in Table 1 indicate that parsers using predicted part-of-speech tags are only slightly worse than the parsers using gold tags (about 2-3% absolute), showing that these methods are robust to tagging errors.

4 Multi-Source Transfer

The previous section focused on transferring an English parser to a new target language. However, there are over 20 treebanks available for a variety of language groups including Indo-European, Altaic (including Japanese), Semitic, and Sino-Tibetan. Many of these are even in standardized formats (Buchholz and Marsi, 2006; Nivre et al., 2007). Past studies have shown that for both part-of-speech tagging and grammar induction, learning with multiple comparable languages leads to improvements (Cohen and Smith, 2009; Snyder et al., 2009; Berg-Kirkpatrick and Klein, 2010). In this section we ex-

		Source Training Language								
		da	de	el	en	es	it	nl	pt	sv
Target Test Language	da	**79.2**	45.2	44.0	45.9	45.0	48.6	46.1	48.1	47.8
	de	34.3	**83.9**	53.2	47.2	45.8	53.4	55.8	55.5	46.2
	el	33.3	52.5	**77.5**	63.9	41.6	59.3	57.3	58.6	47.5
	en	34.4	37.9	45.7	**82.5**	28.5	38.6	43.7	42.3	43.7
	es	38.1	49.4	57.3	53.3	**79.7**	68.4	51.2	66.7	41.4
	it	44.8	56.7	66.8	57.7	64.7	**79.3**	57.6	69.1	50.9
	nl	38.7	43.7	62.1	60.8	40.9	50.4	**73.6**	58.5	44.2
	pt	42.5	52.0	66.6	69.2	68.5	74.7	67.1	**84.6**	52.1
	sv	44.5	57.0	57.8	58.3	46.3	53.4	54.5	66.8	**84.8**

Table 2: UAS for all source-target language pairs. Each column represents which source language was used to train a delexicalized parser and each row represents which target language test data was used. Bold numbers are when source equals target and underlined numbers are the single best UAS for a target language. Results are for all sentence lengths without punctuation.

amine whether this is also true for parser transfer.

Table 2 shows the matrix of source-target language UAS for all nine languages we consider (the original eight target languages plus English). We can see that there is a wide range from 33.3% to 74.7%. There is also a wide range of values depending on the source training data and/or target testing data, e.g., Portuguese as a source tends to parse target languages much better than Danish, and is also more amenable as a target testing language. Some of these variations are expected, e.g., the Romance languages (Spanish, Italian and Portuguese) tend to transfer well to one another. However, some are unexpected, e.g., Greek being the best source language for Dutch, as well as German being one of the worst. This is almost certainly due to different annotation schemes across treebanks. Overall, Table 2 does indicate that there are possible gains in accuracy through the inclusion of additional languages.

In order to take advantage of treebanks in multiple languages, our multi-source system simply concatenates the training data from all non-target languages. In other words, the multi-source direct transfer parser for Danish will be trained by first concatenating the training corpora of the remaining eight languages, training a delexicalized parser on this data and then directly using this parser to analyze the Danish test data. For the multi-source projected parser, the procedure is identical to that in Section 3.2 except that we use the multi-source direct transfer model to seed the algorithm instead of the English-only direct transfer model. For these experiments we still only use English-target parallel data because that is the format of the readily available data in the Europarl corpus.

Table 3 presents four sets of results. The first (best-source) is the direct transfer results for the oracle single-best source language per target language. The second (avg-source) is the mean UAS over all source languages per target language. The third (multi-dir.) is the multi-source direct transfer system. The fourth and final result set (multi-proj.) is the multi-source projected system. The resulting parsers are typically much more accurate than the English direct transfer system (Table 1). On average, the multi-source direct transfer system reduces errors by 10% relative over the English-only direct transfer system. These improvements are not consistent. For Greek and Dutch we see significant losses relative to the English-only system. An inspection of Table 2 shows that for these two languages English is a particularly good source training language.

For the multi-source projected system the results are mixed. Some languages see basically no change relative the multi-source direct transfer model, while some languages see modest to significant increases. But again, there is an overall trend to better models. In particular, starting with an English-only direct transfer parser with 57.0% UAS on average, by adding parallel corpora and multiple source languages we finish with parser having 63.8% UAS on average, which is a relative reduction in error of roughly 16% and more than doubles the performance of a DMV model (Table 1).

Interestingly, the multi-source systems provide, on average, accuracies near that of the single-best source language and significantly better than the average source UAS. Thus, even this simple method of

	best-source		avg-source	gold-POS		pred-POS	
	source	gold-POS	gold-POS	multi-dir.	multi-proj.	multi-dir.	multi-proj.
da	it	48.6	46.3	48.9	49.5	46.2	47.5
de	nl	55.8	48.9	56.7	56.6	51.7	52.0
el	en	63.9	51.7	60.1	65.1	58.5	63.0
es	it	68.4	53.2	64.2	64.5	55.6	56.5
it	pt	69.1	58.5	64.1	65.0	56.8	58.9
nl	el	62.1	49.9	55.8	65.7	54.3	64.4
pt	it	74.8	61.6	74.0	75.6	67.7	70.3
sv	pt	66.8	54.8	65.3	68.0	58.3	62.1
avg		63.7	51.6	61.1	63.8	56.1	59.3

Table 3: UAS for multi-source direct (multi-dir.) and projected (multi-proj.) transfer systems. *best-source* is the best source model from the languages in Table 2 (excluding the target language). *avg-source* is the mean UAS over the source models for the target (excluding target language).

multi-source transfer already provides strong performance gains. We expect that more principled techniques will lead to further improvements. For example, recent work by Søgaard (2011) explores data set sub-sampling methods. Unlike our work, Søgaard found that simply concatenating all the data led to degradation in performance. Cohen et al. (2011) explores the idea learning language specific mixture coefficients for models trained independently on the target language treebanks. However, their results show that this method often did not significantly outperform uniform mixing.

5 Comparison

Comparing unsupervised and parser projection systems is difficult as many publications use non-overlapping sets of languages or different evaluation criteria. We compare to the following three systems that do not augment the treebanks and report results for some of the languages that we considered:

- **USR**: The weakly supervised system of Naseem et al. (2010), in which manually defined universal syntactic rules (USR) are used to constrain a probabilistic Bayesian model. In addition to their original results, we also report results using the same part-of-speech tagset as the systems described in this paper (**USR**†). This is useful for two reasons. First, it makes the comparison more direct. Second, we can generate USR results for all eight languages and not just for the languages that they report.

- **PGI**: The phylogenetic grammar induction (PGI) model of Berg-Kirkpatrick and Klein (2010), in which the parameters of completely unsupervised DMV models for multiple languages are coupled via a phylogenetic prior.

- **PR**: The posterior regularization (PR) approach of Ganchev et al. (2009), in which a supervised English parser is used to generate constraints that are projected using a parallel corpus and used to regularize a target language parser. We report results without treebank specific rules.

Table 4 gives results comparing the models presented in this work to those three systems. For this comparison we use sentences of length 10 or less after punctuation has been removed in order to be consistent with reported results. The overall trends carry over from the full treebank setting to this reduced sentence length setup: the projected models outperform the direct transfer models and multi-source transfer gives higher accuracy than transferring only from English. Most previous work has assumed gold part-of-speech tags, but as the code for USR is publicly available we were able to train it using the same projected part-of-speech tags used in our models. These results are also given in Table 4 under USR†. Again, we can see that the multi-source systems (both direct and projected) significantly outperform the unsupervised models.

It is not surprising that a parser transferred from annotated resources does significantly better than unsupervised systems since it has much more information from which to learn. The PR system of Ganchev et al. (2009) is similar to ours as it also projects syntax across parallel corpora. For Spanish we can see that the multi-source direct transfer parser is better (75.1% versus 70.6%), and this is also true for the multi-source projected parser

	⟵ gold-POS ⟶								← pred-POS →		
	en-dir.	en-proj.	multi-dir.	multi-proj.	USR†	USR	PGI	PR	multi-dir.	multi-proj.	USR†
da	53.2	57.4	58.4	58.8	55.1	51.9	41.6		54.9	54.6	41.7
de	65.9	67.0	74.9	72.0	60.0				63.7	63.4	55.1
el	73.9	73.9	73.5	78.7	60.3				65.2	74.3	53.4
es	58.0	62.3	75.1	73.2	68.3	67.2	58.4	70.6	59.1	56.8	43.3
it	65.5	69.9	75.5	75.5	47.9				65.5	70.2	41.4
nl	67.6	72.2	58.8	70.7	44.0		45.1		56.3	67.2	38.8
pt	77.9	80.6	81.1	86.2	70.9	71.5	63.0		74.0	79.2	66.4
sv	70.4	71.3	76.0	77.6	52.6		58.3		72.0	73.9	59.4
avg	66.6	69.4	71.7	74.1	57.4				63.9	67.5	49.9

Table 4: UAS on sentences of length 10 or less without punctuation, comparing the systems presented in this work to three representative systems from related work. en-dir./en-proj. are the direct/projected English parsers and multi-dir./multi-proj. are the multi-source direct/projected parsers. Section 5 contains a description of the baseline systems.

(73.2%). Ganchev et al. also report results for Bulgarian. We trained a multi-source direct transfer parser for Bulgarian which obtained a score of 72.8% versus 67.8% for the PR system. If we only use English as a source language, as in Ganchev et al., the English direct transfer model achieves 66.1% on Bulgarian and 69.3% on Spanish versus 67.8% and 70.6% for PR. In this setting the English projected model gets 72.0% on Spanish. Thus, under identical conditions the direct transfer model obtains accuracies comparable to PR.[6]

Another projection based system is that of Smith and Eisner (2009), who report results for German (68.5%) and Spanish (64.8%) on sentences of length 15 and less inclusive of punctuation. Smith and Eisner use custom splits of the data and modify a subset of the dependencies. The multi-source projected parser obtains 71.9% for German and 67.8% for Spanish on this setup.[7] If we cherry-pick the source language the results can improve, e.g., for Spanish we can obtain 71.7% and 70.8% by directly transferring parsers form Italian or Portuguese respectively.

6 Discussion

One fundamental point the above experiments illustrate is that even for languages for which no resources exist, simple methods for transferring parsers work remarkably well. In particular, if one can transfer part-of-speech tags, then a large part of transferring unlabeled dependencies has been solved. This observation should lead to a new baseline in unsupervised and projected grammar induction – the UAS of a delexicalized English parser. Of course, our experiments focus strictly on Indo-European languages. Preliminary experiments for Arabic (ar), Chinese (zh), and Japanese (ja) suggest similar direct transfer methods are applicable. For example, on the CoNLL test sets, a DMV model obtains UAS of 28.7/41.8/34.6% for ar/zh/ja respectively, whereas an English direct transfer parser obtains 32.1/53.8/32.2% and a multi-source direct transfer parser obtains 39.9/41.7/43.3%. In this setting only Indo-European languages are used as source data. Thus, even across language groups direct transfer is a reasonable baseline. However, this is not necessary as treebanks are available for a number of language groups, e.g., Indo-European, Altaic, Semitic, and Sino-Tibetan.

The second fundamental observation is that when available, multiple sources should be used. Even through naive multi-source methods (concatenating data), it is possible to build a system that has comparable accuracy to the single-best source for all languages. This advantage does not come simply from having more data. In fact, if we randomly sampled from the multi-source data until the training set size was equivalent to the size of the English data, then the results still hold (and in fact go up slightly for some languages). This suggests that even better transfer models can be produced by separately weighting each of the sources depending on the target language – either weighting by hand, if we know the language group of the target language, or auto-

[6] Note that the last set of results was obtained by using the same English training data as Ganchev et al. Using the CoNLL 2007 English data set for training, the English direct transfer model is 63.2% for Bulgarian and 58.0% for Spanish versus 67.8% and 70.6% for PR, highlighting the large impact that difference treebank annotation standards can have.

[7] Data sets and evaluation criteria obtained via communications with David Smith and Jason Eisner.

matically, if we do not. As previously mentioned, the latter has been explored in both Søgaard (2011) and Cohen et al. (2011).

7 Conclusions

We presented a simple, yet effective approach for projecting parsers from languages with labeled training data to languages without any labeled training data. Central to our approach is the idea of delexicalizing the models, which combined with a standardized part-of-speech tagset allows us to directly transfer models between languages. We then use a constraint driven learning algorithm to adapt the transferred parsers to the respective target language, obtaining an additional 16% error reduction on average in a multi-source setting. Our final parsers achieve state-of-the-art accuracies on eight Indo-European languages, significantly outperforming previous unsupervised and projected systems.

Acknowledgements: We would like to thank Kuzman Ganchev, Valentin Spitkovsky and Dipanjan Das for numerous discussions on this topic and comments on earlier drafts of this paper. We would also like to thank Shay Cohen, Dipanjan Das, Noah Smith and Anders Søgaard for sharing early drafts of their recent related work.

References

T. Berg-Kirkpatrick and D. Klein. 2010. Phylogenetic grammar induction. In *Proc. of ACL*.

P. Blunsom and T. Cohn. 2010. Unsupervised induction of tree substitution grammars for dependency parsing. *Proc. of EMNLP*.

P. F. Brown, V. J. Della Pietra, S. A. Della Pietra, and R. L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19.

S. Buchholz and E. Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proc. of CoNLL*.

D. Burkett, S. Petrov, J. Blitzer, and D. Klein. 2010. Learning better monolingual models with unannotated bilingual text. In *Proc. of CoNLL*.

G. Carroll and E. Charniak. 1992. Two experiments on learning probabilistic dependency grammars from corpora. In *Proc. of the Working Notes of the Workshop Statistically-Based NLP Techniques*.

M.W. Chang, L. Ratinov, and D. Roth. 2007. Guiding semi-supervision with constraint-driven learning. In *Proc. of ACL*.

M. Chang, D. Goldwasser, D. Roth, and V. Srikumar. 2010. Structured output learning with indirect supervision. In *Proc. of ICML*.

E. Charniak. 2000. A maximum-entropy-inspired parser. In *Proc. of NAACL*.

S. Clark and J. R. Curran. 2004. Parsing the WSJ using CCG and log-linear models. In *Proc. of ACL*.

S.B. Cohen and N.A. Smith. 2009. Shared logistic normal distributions for soft parameter tying in unsupervised grammar induction. In *Proc. of NAACL*.

S.B. Cohen, D. Das, and N.A. Smith. 2011. Unsupervised structure prediction with non-parallel multilingual guidance. In *Proc. of EMNLP*.

M. Collins, J. Hajič, L. Ramshaw, and C. Tillmann. 1999. A statistical parser for Czech. In *Proc. of ACL*.

M. Collins. 1997. Three generative, lexicalised models for statistical parsing. In *Proc. of ACL*.

M. Collins. 2002. Discriminative training methods for hidden markov models: Theory and experiments with perceptron algorithms. In *Proc. of ACL*.

D. Das and S. Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proc. of ACL-HLT*.

K. Ganchev, J. Gillenwater, and B. Taskar. 2009. Dependency grammar induction via bitext projection constraints. In *Proc. of ACL-IJCNLP*.

K. Ganchev, J. Graça, J. Gillenwater, and B. Taskar. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*.

D. Gildea. 2001. Corpus variation and parser performance. In *Proc of EMNLP*.

K. Hall, R. McDonald, J. Katz-Brown, and M. Ringgaard. 2011. Training dependency parsers by jointly optimizing multiple objectives. In *Proc. of EMNLP*.

R. Hwa, P. Resnik, A. Weinberg, C. Cabezas, and O. Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Natural Language Engineering*, 11(03):311–325.

D. Klein and C. D. Manning. 2004. Corpus-based induction of syntactic structure: models of dependency and constituency. In *Proc. of ACL*.

P. Koehn. 2005. Europarl: A parallel corpus for statistical machine translation. In *MT Summit*.

M. P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn treebank. *Computational Linguistics*, 19.

D. McClosky, E. Charniak, and M. Johnson. 2006. Reranking and self-training for parser adaptation. In *Proc. of ACL*.

R. McDonald, K. Crammer, and F. Pereira. 2005. Online large-margin training of dependency parsers. In *Proc. of ACL*.

T. Naseem, H. Chen, R. Barzilay, and M. Johnson. 2010. Using universal linguistic knowledge to guide grammar induction. In *Proc. of EMNLP*.

J. Nivre and J. Nilsson. 2005. Pseudo-projective dependency parsing. In *Proc. of ACL*.

J. Nivre, J. Hall, and J. Nilsson. 2006. Maltparser: A data-driven parser-generator for dependency parsing. In *Proc. of LREC*.

J. Nivre, J. Hall, S. Kübler, R. McDonald, J. Nilsson, S. Riedel, and D. Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proc. of EMNLP-CoNLL*.

J. Nivre. 2008. Algorithms for deterministic incremental dependency parsing. *Computational Linguistics*, 34(4):513–553.

S. Petrov, L. Barrett, R. Thibaux, and D. Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *Proc. of ACL*.

S. Petrov, P. Chang, M. Ringgaard, and H. Alshawi. 2010. Uptraining for accurate deterministic question parsing. In *EMNLP '10*.

S. Petrov, D. Das, and R. McDonald. 2011. A universal part-of-speech tagset. In *ArXiv:1104.2086*.

Y. Seginer. 2007. Fast unsupervised incremental parsing. In *Proc. of ACL*.

L. Shen and A.K. Joshi. 2008. Ltag dependency parsing with bidirectional incremental construction. In *Proc. of EMNLP*.

N.A. Smith and J. Eisner. 2005. Contrastive estimation: Training log-linear models on unlabeled data. In *Proc. of ACL*.

D.A. Smith and J. Eisner. 2009. Parser adaptation and projection with quasi-synchronous grammar features. In *Proc. of EMNLP*.

D.A. Smith and N.A. Smith. 2004. Bilingual parsing with factored estimation: Using english to parse korean. In *Proc. of EMNLP*.

B. Snyder, T. Naseem, J. Eisenstein, and R. Barzilay. 2009. Adding more languages improves unsupervised multilingual part-of-speech tagging: A Bayesian nonparametric approach. In *Proc. of NAACL*.

A. Søgaard. 2011. Data point selection for cross-language adaptation of dependency parsers. In *Proc. ACL*.

V.I. Spitkovsky, H. Alshawi, and D. Jurafsky. 2010. From baby steps to leapfrog: How "less is more" in unsupervised dependency parsing. In *Proc. of NAACL-HLT*.

S. Vogel, H. Ney, and C. Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proc. of COLING*.

W. Wang and M. P. Harper. 2004. A statistical constraint dependency grammar (CDG) parser. In *Proc. of the Workshop on Incremental Parsing: Bringing Engineering and Cognition Together*.

D. Zeman and P. Resnik. 2008. Cross-language parser adaptation between related languages. In *NLP for Less Privileged Languages*.

Y. Zhang and S. Clark. 2008. A Tale of Two Parsers: Investigating and Combining Graph-based and Transition-based Dependency Parsing. In *Proc. of EMNLP*.