Zero-resource Dependency Parsing: Boosting Delexicalized Cross-lingual Transfer with Linguistic Knowledge

Lauriane Aufrant[1,2] and **Guillaume Wisniewski**[1] and **François Yvon**[1]
[1]LIMSI, CNRS, Univ. Paris-Sud, Université Paris-Saclay, 91 405 Orsay, France
[2]DGA, 60 boulevard du Général Martial Valin, 75 509 Paris, France
{lauriane.aufrant,guillaume.wisniewski,francois.yvon}@limsi.fr

Abstract

This paper studies cross-lingual transfer for dependency parsing, focusing on very low-resource settings where delexicalized transfer is the only fully automatic option. We show how to boost parsing performance by rewriting the source sentences so as to better match the linguistic regularities of the target language. We contrast a data-driven approach with an approach relying on linguistically motivated rules automatically extracted from the World Atlas of Language Structures. Our findings are backed up by experiments involving 40 languages. They show that both approaches greatly outperform the baseline, the knowledge-driven method yielding the best accuracies, with average improvements of +2.9 UAS, and up to +90 UAS (absolute) on some frequent PoS configurations.

1 Introduction

The need to automatically process an increasing number of languages has made obvious the extreme dependency of standard development pipelines on in-domain, annotated resources that are required to train efficient statistical models. However, for most languages, annotated corpora only exist for a restricted number of domains, when they exist at all. In response to such low-resource scenario, four main strategies have been considered. The first is to hire experts and handcraft these resources, possibly with the help of active learning techniques: Garrette and Baldridge (2013) show that this strategy can be effective and probably cheaper than expected. An alternative is to use models learned on some resource-rich *source language(s)* to process a low-resource *target language*; note that this is only possible once the source and target data have been mapped into a shared representation space (Zeman and Resnik, 2008). When source-target parallel corpora are available, a third approach projects annotations across languages via alignment links (Yarowsky and Ngai, 2001; Hwa et al., 2005; Lacroix et al., 2016). A variant using *artificial* parallel corpora, obtained via Machine Translation, is suggested and discussed by Tiedemann et al. (2014).

In this work, we focus on the problem of learning dependency parsers for an under-resourced language and consider the *delexicalized transfer* approach of Zeman and Resnik (2008), in which the shared source-target representation is obtained by replacing all tokens by their PoS (assuming a common tagset). Thanks to this language-independent representation, a model trained with annotated sentences in a source language can be readily applied to parse sentences in any other language. Delexicalized techniques are especially useful in very low-resource settings, in which existing parallel corpora are likely to be too small or even non-existing. The development of cross-linguistically homogeneous and consistent schemes for PoS labels (Petrov et al., 2012) and, more recently, for dependency trees (McDonald et al., 2013) has been of great help to improve the applicability and effectiveness of delexicalized transfer methods. We contend, however, that having a universal PoS inventory is only a first step towards making the source and target languages more alike. In particular, these shared representations may hide fundamental differences in word order between source and target languages. As explained in § 2, these

divergences introduce systematic biases in parsers: since many features rely on word linear sequence, their distribution across languages varies in great proportions, preventing useful generalizations to be effectively transferred cross-linguistically.

In the remaining sections, we study ways to improve the performance of delexicalized techniques by making the source word sequence more similar to target sentences, prior to transferring information. Two extensions are contrasted: a data-driven approach and a knowledge-driven approach (§ 3). The former uses PoS-based statistical language models estimated on target data while the latter relies only on the World Atlas of Language Structures (WALS) (Dryer and Haspelmath, 2013), which contains a series of linguistic typological features documenting 2,679 languages. Experiments on 40 languages exhibiting very different characteristics and covering several language families show that both methods outperform standard delexicalized transfer by a wide margin (§ 4), with the knowledge-based approach having the additional benefit to even dispense with the need of unlabeled target data and consequently to be readily usable for more than thousand languages. Incidentally, our experiments thoroughly re-evaluates previous proposals for improving baseline delexicalized transfer.

2 Motivations

2.1 Principles of Transition-Based Dependency Parsing

Transition-based dependency parsers (Nivre, 2008) are among the most popular methods for computing a syntactic structure. For clarity, we illustrate our work on greedy ARCEAGER parsers which have achieved state-of-the-art performance for many languages. However, our motivations hold regardless of the chosen parsing system, and exploratory experiments with our methods have shown similar improvements with other parsers (including graph-based parsers).

In an ARCEAGER parser, the parse tree is built incrementally while traversing the sentence from left to right, by executing elementary actions that either move words in a buffer and a stack (via SHIFT and REDUCE actions) or create dependency relationships between the word on top of the stack and the leftmost word in the buffer (using the LEFT or RIGHT actions depending whether the head is in the buffer or on the stack).

The actions performed during parsing are predicted by a feature-based classifier, a common choice being the averaged perceptron of Collins and Roark (2004). It is custom to base the classifier decisions on a limited window centered on the two tokens which could be moved or attached; the following features[1] are typically extracted from these neighborhoods and combined together to yield feature tuples: top of the stack (generally denoted s_0) and deeper stack elements (s_1, s_2) to its left, head of the buffer (n_0) and additional tokens (n_1, n_2) on its right.

Transition-based parsers heavily rely on word order: for instance, as shown in Figure 1, in an ARCEAGER system, the dependency between two words will be predicted by two different actions depending whether the head occurs before or after its dependent. More importantly, most features used in a dependency parser (no matter the transition system) are sensitive to the word order, as they encode the position of the word in the stack or in the buffer which, in turn, depends on the position of the word in the sentence.[2]

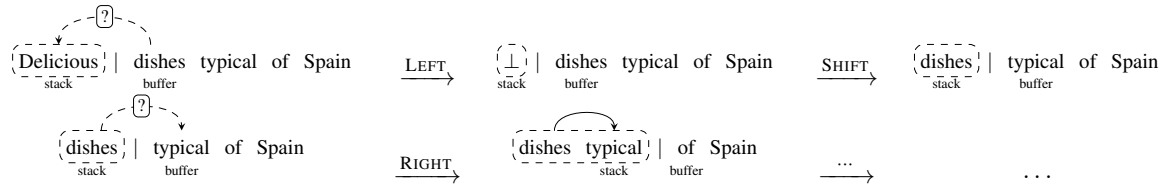

Figure 1: An order-sensitive sequence of transitions computing a dependency tree.

[1] For lexicalized parsers: the word forms and the PoS, for delexicalized: only the PoS.

[2] Graph-based parsing with standard feature templates is slightly less order-dependent, since the classification task and the features of the candidate dependent and head are already abstracted from the linear sequence. However, many features, related for instance to words located *between* these two tokens, remain sensitive to word order and our statement still holds.

2.2 A Waste of Cross-Lingual Knowledge

Delexicalized transfer has proven to be an effective method to transfer parsers between languages (Zeman and Resnik, 2008; McDonald et al., 2013). However, while delexicalized transfer extracts useful language-independent knowledge from training instances in the source language, we claim that this knowledge is often not encoded in the right form to be effectively used to process target sentences, due to divergences in word ordering.[3]

We illustrate this on delexicalized transfer from English to French. Let us assume that we have a delexicalized English parser that is able to perfectly predict the dependency structure of the noun phrase *the following question* and we use it to annotate the corresponding French phrase *la question suivante* (literally, *the question following*[4]). Thanks to recent efforts in defining universal annotation schemes for syntactic information, notably the Universal Dependencies (UD) project (Nivre et al., 2016), these phrases can be represented in a unified manner by mapping word forms into the corresponding PoS, yielding respectively DET ADJ NOUN and DET NOUN ADJ. As the English parser has learned that 'DETs depend on NOUNs' and that 'ADJs depend on NOUNs', the appropriate parse for the French phrase should be obvious, as these rules apply cross-linguistically. PoS sequences thus seem to provide an appropriate level of abstraction for cross-lingual transfer.

However, contrary to what this intuition suggests, the transfer of the ADJ-NOUN dependency often fails in practice. This is because the features underlying the high-level rules stated above are in fact order-dependent. Indeed, when parsing the French phrase, the parser configuration will be mainly described by the feature pair 's_0=NOUN $\wedge$ n_0=ADJ' (as *question* appears before *suivante*, it will be put on the stack first) while for the English phrase the relevant parser configuration would look like 's_0=ADJ $\wedge$ n_0=NOUN'. For lack of connecting these two situations, the parser has no way to predict the correct attachment in French using only English training instances.

Experimentally,[5] and denoting $\mathrm{UAS}\left[\begin{smallmatrix}C_2\\C_1\end{smallmatrix}\right]$ the percentage of C_1 tokens depending on a C_2 token that are correctly attached by the parser, while the English delexicalized model has a $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\textsc{Adj}\end{smallmatrix}\right]$ of 91.1% on English, it drops down to 50.8% for French. This decrease results directly from the word order difference between French and English, as English adjectives are almost always preposed[6] while their position in French is less deterministic: in the French UD, 28% of the adjectives occur before their head noun and 72% after it. As a result, the $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\textsc{Adj}\end{smallmatrix}\right]$ score on French actually decomposes as 96.8% for $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{preposed }\textsc{Adj}\end{smallmatrix}\right]$ and 34.5% for $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{postposed }\textsc{Adj}\end{smallmatrix}\right]$.[7] These observations highlight the impact of word order on delexicalized transfer: attachment patterns are not robust to variations in word ordering. Note that transfer from French to English is much more successful, with a $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\textsc{Adj}\end{smallmatrix}\right]$ of 80.5%. This is because the source language (here French) contains a sufficiently large number of preposed adjectives, which makes it possible to learn the patterns that are useful for English.

The discrepancies in word order can have an even more dramatic effect when transferring parsers between languages in which adjectives have a fixed position. This, for instance, happens when the source is Bulgarian (almost only preposed adjectives) and the target is Hebrew (only postposed): the resulting $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\textsc{Adj}\end{smallmatrix}\right]$ is as low as 28.7% (compared to an overall UAS of 60.1%). In the reverse direction, it drops down to 2.8% ($\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{preposed }\textsc{Adj}\end{smallmatrix}\right]$ of 0.7%, $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{postposed }\textsc{Adj}\end{smallmatrix}\right]$ of 54.5%, with an overall UAS of 50.6%).[8]

The impact of differences in word order on cross-lingual transfer is not limited to the attachment of adjectives. Consider, for instance, the English phrase *the neighbor's car* (DET NOUN PART NOUN) and

[3] The issues described in this section are at least partially solved by transfer with annotation projection but these techniques require parallel data that are not always available.

[4] Keeping the original order (*la suivante question*) would be wrong in French.

[5] Our experimental data and protocols are presented in Section 4.

[6] In the English UD corpus, 93% of the adjectives come before the noun they depend on.

[7] This observation is consistent with the English monolingual scores (93.2% for the $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{preposed }\textsc{Adj}\end{smallmatrix}\right]$ majority case, and 55.0% for the much rarer $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\text{postposed }\textsc{Adj}\end{smallmatrix}\right]$ case).

[8] Source data quality cannot be the only cause of such poor results: when delexicalized models apply monolingually, $\mathrm{UAS}\left[\begin{smallmatrix}\textsc{Noun}\\\textsc{Adj}\end{smallmatrix}\right]$ is 97.4% in Bulgarian and 88.4% in Hebrew.

its French translation *la voiture du voisin* (DET NOUN ADP NOUN). After attaching function words, all that remains for the parser to process is the bigram NOUN NOUN: while the English parser has been trained to predict a left dependency (*car* being the head of *neighbor*), for French it must predict a right dependency (*voiture* being the head of *voisin*). Here the discrepancy of genitives' position across languages does not involve unseen features, but still leads the model to predict a wrong dependency with high confidence.

Our work aims at addressing such scenarios in which knowledge transfer is impeded by the word order of the source language. While current state-of-the-art models learn that 'an ADJ followed by a NOUN depends on that NOUN' and 'the first NOUN depends on the second NOUN', we would like them to transfer more abstract patterns such as 'ADJs depend on NOUNs' and 'genitives depend on NOUNs', leaving it up to the target side to decide which of both NOUNs plays the role of genitive.

3 Boosting Delexicalized Transfer

3.1 Reshaping Training Instances

In this work, we propose to preprocess the source data before they are used to train a delexicalized parser, that will then be directly applied on target sentences. This preprocessing aims at making the source word sequences more similar to target sentences, with the goal to make the cross-lingual knowledge more accessible after transfer. The available information is the same before and after preprocessing (no dependency is ever added), but is presented at training time in a form that should make it more useful at test time.

In the following, we introduce two ways of generating such transformations, by removing or permuting tokens. The first approach uses a language model estimated on target PoS sequences to find the most similar word order between the source and target languages in a lattice containing local reorderings of the source sentence. The second strategy relies on a data bank of linguistic typological features, the WALS (Dryer and Haspelmath, 2013), to generate a series of heuristic transformation operations.

The problem of finding good reorderings of a source sentence is closely related to the problem of word (p)reordering in Statistical Machine Translation (SMT) (Bisazza and Federico, 2016). However, where preordering aims to find an optimal (for SMT) permutation of source words *for each source sentence*, our objective is less ambitious, as we only intend to 'fix' a sufficiently large number of divergent patterns between the source and target languages, so as to increase the effectiveness of transfer *at the model level*.

3.2 Optimally Reordering the Training Corpus with a Language Model

Our first resource-light approach consists of two steps. We first generate a small subset of possible token permutations, compactly encoded in a finite-state graph. In our experiments, we consider all the permutations licensed by the MJ-2 reordering scheme (Kumar and Byrne, 2005), which generates all possible local permutations within a window of three words. Machine Translation experiments have shown that the MJ-2 constraints capture lots of plausible reorderings (Dreyer et al., 2007). In the context of cross-lingual transfer, its local nature allows to correct several important divergences in word order (e.g. the adjective-noun divergence described in § 2.2), while keeping the size of the reordering lattice polynomial with respect to the sentence length (Lopez, 2009).

The permutation lattice is then searched for a reordering that (a) corresponds to a high probability target PoS sequence and (b) preserves the projectivity constraint. In practice, we first generate the lattice of MJ-2 reorderings, score it with a language model estimated on the target PoS sequences, and extract the 1,000-best sequences. After filtering non-projective trees, we retain the one-best sequence (if one projective tree exists), or the original sequence otherwise. We expect the word order of this transformed source to be very close to the word order of a typical target sentence. We then transform the gold dependency tree according to this permutation and use it to train a target-adapted model.

This approach can be viewed as an extension of the data selection technique of Søgaard (2011) in which the delexicalized model is trained only on the source examples that are the most *relevant* for the target at hand. The similarity between the source and target languages is based on the similarity between

their PoS sequences: experimentally, the author retains the 90% sentences with lowest perplexity according to a target PoS language model (PoSLM). We add here an extra degree of freedom by allowing changes in the word order of the source PoS sequence, rather than simply discarding sentences.

3.3 Adapting the Training Corpus with Rewrite Rules

Our second proposal takes advantage of the linguistic knowledge that is now available for many languages. We use here the WALS, which contains a series of linguistic features documenting 2,679 languages. Some of these features are of prime interest for our study, and express general properties related to word order. In this work we focus on the following seven features that describe whether some PoS classes exist in a language and their relative position (preposed or postposed to the noun, or no dominant order): 37A (definite articles), 38A (indefinite articles), 85A (order of adposition and noun), 86A (order of genitive and noun), 87A (order of adjective and noun), 88A (order of demonstrative and noun) and 89A (order of numeral and noun).[9]

We first extract the relevant features for each language considered in our study, quantify their value and automatically transform them to relate to the raw PoS sequences found in our corpora. We extrapolate the order of DET and NOUN from feature 88A and identify the genitives mentioned by feature 86A as NOUNs or PROPNs depending on a NOUN. With an otherwise straightforward mapping, this results in the following set of properties: no definite DET, no indefinite DET (including the affix cases), and a precedence rate (denoted PR) of 0% (postposed), 50% (no dominant order) or 100% (preposed) for ADPs (resp. genitives, ADJs, DETs, NUMs) depending on a NOUN.[10]

The 'No dominant order' feature value of WALS provides very useful quantitative information: contrary to the PoSLM-based approach, which puts hard constraints on each phenomenon by choosing a reordering even when several choices would be almost equally likely, WALS features indicate when and how to balance our transformed treebanks.

By comparing two languages based on their feature values, it is then possible to define actionable transformation rules that remove or permute tokens and their associated subtrees. Table 1 lists the transformation rules derived from each pair of features. We preferred smooth transformations (with mean PR objectives and error margins) to prevent a full transformation of the corpus and a risk of information losses if the child position is less deterministic than expected. For instance, in the case of transfer from English (ADP-NOUN) to Japanese (NOUN-ADP) and according to the fourth transformation rule, we target a precedence rate of ADPs to NOUNs between 45% and 55%. This means that during source treebank traversal, while the precedence rate in previous sentences is above 55% (resp. below 45%), any encountered ADP-NOUN (resp. NOUN-ADP) bigram holding a dependency is switched, along with their dependents to preserve projectivity. According to first rule, for transfer to Czech (no definite article) from any source, all definite articles are systematically removed from source data.

Source feature	Target feature	Transformation rule
any	no DEF-DET	remove all definite DETs
any	no IND-DET	remove all indefinite DETs
$\mathrm{PR} = 0\%$	$\mathrm{PR} \geq 50\%$	switch subtrees to reach $\mathrm{PR} = 50\%$ (with 5% error margin)
$\mathrm{PR} = 100\%$	$\mathrm{PR} \leq 50\%$	switch subtrees to reach $\mathrm{PR} = 50\%$ (with 5% error margin)
$\mathrm{PR} = 50\%$	$\mathrm{PR} = 100\%$	switch subtrees to reach $\mathrm{PR} = 75\%$ (with 5% error margin)
$\mathrm{PR} = 50\%$	$\mathrm{PR} = 0\%$	switch subtrees to reach $\mathrm{PR} = 25\%$ (with 5% error margin)

Table 1: Transformation rules extracted from the comparison of the feature values of a language pair. All other feature pairs result in a no-op.

For each sentence, we apply each rule on the whole sequence (and then iterate 3 times to capture recur-

[9] We do not consider here features (81A, 82A, etc.) describing the relative position of a head VERB and its dependents. Their use would require us to condition our preprocessing patterns on labeled dependency relationship in the source, a task we leave for future work.

[10] For ADPs, and for resilience to annotation inconsistencies, we also include ADPs that are heads of NOUNs.

sive phenomena). Such heuristic rule application strategy is undoubtedly sensitive to the rule ordering, but we have not yet investigated this aspect and simply apply rules according to the lexicographic order of the child tag, breaking ties using word position.

In comparison to the PoSLM-based approach, the WALS-based approach suffers from a lack of exhaustivity regarding word order; by design, less phenomena will be captured. However, since the objective is not the best possible reordering but only more compatible PoS sequences, exhaustivity is probably not a big issue. Besides, working with a discrete and reduced set of transformation operations gives us a better control on the rewriting of dependencies. It also allows us to use extra operations such as word deletion, a transformation that may be difficult to control in the approach described in § 3.2.

Altogether, this linguistically rich method presents a notable upside: since all the required information is available in WALS, it is readily usable for more than thousand languages. Provided that PoS tags can be generated for the target data to parse, no extra resource is required, while estimating a PoSLM requires a sufficiently large corpus of reliably PoS-tagged target data.

4 Experiments

4.1 Experimental Setup

We evaluate our proposals on the Universal Dependencies corpus[11] (Nivre et al., 2016) and compare them with three baselines: (a) standard delexicalized transfer, (b) the data point selection method of Søgaard (2011) and (c) the weighted multi-source combination of Rosa and Zabokrtsky (2015), that weights and combines the hypotheses of several delexicalized source models using KL_{cpos^3} (Kullback-Leibler divergence of coarse PoS trigram distributions) as a syntactic similarity metric between languages. We also include the UAS of KL_{cpos^3} multi-source combination built on top of our knowledge-based model.

In all our experiments, we consider 3-gram PoS language models estimated on the training sets of UD. The KL_{cpos^3} metric is estimated on the same PoS sequences. From WALS, we extract and use the 37A, 38A and 85A to 89A features. For some languages, this information was incomplete. We completed missing features with a majority vote of the languages of same genus if available in the database; otherwise (i.e. for ancient languages, all absent from WALS) we assumed that there were separate article tokens and that there was no dominant order for word order features.

For each component of the algorithms, we use the universal PoS tagset and gold PoS tags. While this scenario is probably unrealistic, it allows us to get a clearer picture of the net effect of a better syntactic knowledge transfer, since possible sources of discrepancies between languages (e.g. more or less noisy tag labels) have been removed. The parser is a greedy ARCEAGER transition-based parser trained with a dynamic oracle (Goldberg and Nivre, 2012), an averaged perceptron classifier (Collins and Roark, 2004) and Zhang and Nivre (2011)'s feature templates (assuming fully delexicalized representations and unlabeled arcs). We use the PanParser implementation (Aufrant and Wisniewski, 2016) and all the code used in this work is available at `https://perso.limsi.fr/aufrant/`.

4.2 Results

Table 2 presents UAS results for the various transfer methods considered. As these experiments amount to 6,320 evaluated parsers, we provide the results in a compacted form as follows. For each target language, for mono-source transfer, we report the scores of the worst, median, best sources and the average score (or average gain) over all sources.

Overall, both preprocessing techniques outperform the direct transfer method of Zeman and Resnik (2008) as well as the selection strategy of Søgaard (2011). The WALS-based rewriting approach yields higher improvements (+2.9 UAS on average) than the PoSLM-based reordering strategy (+2.3 UAS on average). Thanks to the variety and the large number of sources, the multi-source methods have here much higher accuracies, often better than the best source; even in this setting, using WALS provides us with a slight improvement over the baseline multi-source parser.

[11] We consider the version 1.3 of the UD treebank. In order to present only fair sources, for languages where several treebanks are available, we retain only the *main* treebank. This is the case for the following languages: cs, en, es, fi, grc, la, nl, pt, ru, sl and sv.

Target	[Mono-source																] [Multi-source]	
	Delexicalized				PoSLM selection				PoSLM reordering				WALS rewrite rules				Delex.	WALS
	min	med	max	avg	min	med	max	Δavg	min	med	max	Δavg	min	med	max	Δavg		
ar	5.1	43.2	56.9	36.1	4.8	43.0	57.2	-0.2	18.9	45.1	57.2	+5.6	12.2	47.6	56.9	**+5.7**	57.3	57.8
bg	26.4	67.5	78.9	59.6	26.3	67.5	78.9	-0.1	35.5	65.1	74.4	+1.5	27.2	67.6	78.9	**+1.8**	79.6	79.0
ca	28.4	62.3	78.5	57.8	27.9	62.0	78.7	-0.1	33.5	60.8	75.5	-0.2	30.4	66.2	78.6	**+1.9**	79.2	79.1
cs	29.7	58.5	74.0	54.3	29.2	58.6	74.0	-0.0	37.1	58.4	68.8	**+1.4**	30.8	59.3	73.8	+1.4	74.0	73.8
cu	22.6	58.5	74.7	53.9	22.6	58.7	75.4	+0.0	38.2	60.2	70.0	**+3.8**	24.3	60.3	74.7	+1.6	74.7	74.7
da	28.0	64.5	75.3	58.2	27.5	63.8	75.3	-0.2	40.3	61.0	70.4	-0.0	28.6	64.6	74.5	**+1.4**	75.7	75.2
de	36.2	61.2	70.5	57.7	35.9	61.0	70.0	-0.2	45.4	61.1	69.3	+1.3	43.0	61.8	70.8	**+1.5**	68.7	68.7
el	29.0	51.0	67.8	49.5	28.9	50.5	68.2	-0.0	33.5	51.6	64.9	+0.4	29.8	51.6	67.6	**+1.7**	67.0	66.2
en	33.1	56.5	65.8	52.8	32.5	56.2	65.5	-0.2	38.4	57.0	63.8	**+1.2**	32.2	58.4	65.9	+1.2	65.7	66.0
es	30.0	63.9	78.5	58.9	29.3	64.4	78.5	-0.0	37.6	62.8	76.1	+0.3	31.5	66.6	78.4	**+1.8**	79.2	79.3
et	28.4	53.1	69.4	51.5	26.9	52.9	69.6	-0.2	36.3	57.0	67.9	+3.7	37.1	57.9	69.3	**+4.4**	69.4	69.3
eu	20.7	45.2	57.8	44.2	20.9	46.4	57.7	-0.1	24.3	54.6	64.1	**+8.1**	24.6	52.0	63.3	+5.7	55.7	60.9
fa	17.9	45.3	56.1	40.3	17.6	45.0	56.0	-0.1	26.7	46.3	56.8	+3.2	25.5	48.4	58.8	**+5.2**	61.4	63.3
fi	27.4	48.1	62.1	46.6	27.4	48.2	61.8	-0.0	32.4	50.2	58.8	+2.3	32.4	53.0	62.2	**+3.7**	62.1	62.2
fr	30.9	64.0	79.1	59.0	29.9	63.9	78.7	-0.2	35.0	61.4	76.8	+0.5	34.2	66.0	78.9	**+1.8**	79.8	79.5
ga	16.4	56.0	65.8	50.1	16.3	56.2	66.4	-0.1	26.6	56.2	64.6	+1.8	20.8	59.0	65.3	**+2.3**	67.4	67.2
gl	33.0	40.3	47.5	40.6	32.8	40.5	47.5	-0.1	32.1	43.0	48.2	+1.5	35.6	43.7	51.0	**+2.6**	46.7	46.6
got	26.4	58.0	72.7	54.3	26.4	57.5	73.4	-0.0	38.0	60.0	66.3	**+3.1**	28.2	58.9	73.9	+0.9	72.7	73.9
grc	32.5	53.9	57.3	50.7	29.8	53.9	57.8	-0.1	39.0	53.9	58.3	**+1.1**	32.4	53.9	57.8	+0.0	61.0	60.3
he	20.1	53.8	68.0	49.9	19.8	54.2	67.7	+0.1	30.2	54.1	63.6	+1.2	21.9	55.4	65.8	**+1.6**	71.2	68.7
hi	11.0	27.1	66.5	32.3	11.1	26.9	65.8	-0.1	22.0	37.5	61.6	**+6.7**	19.8	33.8	66.8	+5.4	37.1	44.2
hr	26.8	55.4	71.2	52.0	26.0	56.3	70.9	-0.2	35.7	56.3	66.9	**+1.7**	28.9	56.3	70.3	+1.5	73.9	73.0
hu	27.8	52.7	67.8	50.8	27.1	53.1	68.2	-0.0	40.4	56.3	65.4	**+4.4**	40.1	55.4	68.3	+3.4	63.0	64.4
id	17.4	49.2	70.1	48.8	18.2	49.0	70.3	+0.1	27.6	50.2	66.1	+1.5	23.1	53.8	69.6	**+3.7**	70.8	71.9
it	31.0	67.1	82.6	61.7	30.4	66.9	82.2	-0.1	38.1	67.0	80.7	+1.2	34.0	70.8	82.3	**+2.2**	83.2	82.9
ja	7.0	18.6	72.6	26.7	7.2	18.3	72.2	+0.1	15.7	32.9	70.6	+10.0	18.1	35.2	72.3	**+11.4**	63.3	63.5
kk	10.7	33.0	56.3	32.4	10.9	34.0	54.3	+0.2	17.9	35.2	52.6	+3.4	20.6	38.5	55.6	**+4.9**	53.9	54.6
la	14.4	49.9	64.1	47.1	14.3	50.0	63.5	+0.0	19.5	51.4	61.8	+1.9	21.1	53.3	63.3	**+2.1**	58.3	60.0
lv	22.6	40.3	55.7	40.6	22.5	40.1	55.8	-0.2	28.7	42.6	51.0	+1.8	35.0	47.1	55.4	**+5.5**	50.2	57.3
nl	27.5	51.9	61.7	48.9	27.9	52.2	62.2	+0.1	32.4	49.3	56.8	-2.4	28.4	52.4	60.4	**+0.5**	62.3	60.7
no	25.8	64.0	76.6	57.1	25.6	63.9	76.7	-0.1	37.2	58.5	69.2	-0.2	26.5	63.8	76.2	**+1.3**	76.6	76.5
pl	25.7	62.1	77.9	59.2	25.6	62.7	77.9	-0.1	36.0	65.6	76.2	**+3.4**	29.5	65.5	77.4	+2.4	78.0	77.6
pt	30.8	62.8	75.5	56.7	30.2	63.3	75.5	+0.0	35.7	60.0	73.5	-0.9	32.8	63.5	75.4	**+1.4**	75.5	75.7
ro	19.8	55.5	69.2	51.8	18.6	55.7	68.7	-0.1	31.7	58.5	67.7	+3.1	24.1	60.5	70.3	**+4.0**	71.8	72.0
ru	26.8	53.9	69.0	51.3	26.1	54.0	68.9	-0.0	34.6	55.1	67.9	+2.3	30.9	59.2	68.7	**+4.2**	71.0	70.4
sl	30.6	65.2	80.4	59.4	30.4	65.1	80.5	-0.0	41.8	64.8	77.4	**+2.7**	30.5	64.9	80.4	+1.4	80.4	80.4
sv	29.4	62.7	75.5	56.9	29.4	62.3	75.5	-0.2	39.6	60.1	70.6	+0.5	30.4	63.9	74.9	**+2.3**	73.1	73.5
ta	9.1	36.3	66.3	36.8	9.1	35.6	66.4	-0.0	18.9	43.7	64.5	**+6.2**	19.0	41.1	65.8	+4.7	66.3	65.8
tr	14.1	35.3	67.0	38.8	14.7	35.4	67.0	-0.1	19.5	39.5	64.5	+2.0	21.5	40.8	67.8	**+3.5**	58.6	58.5
zh	15.6	32.5	43.1	31.7	15.8	32.5	43.0	+0.1	18.9	35.5	41.8	+2.3	20.1	36.6	44.1	**+3.5**	40.2	42.2
Avg	23.7	52.0	68.2	49.2	23.3	52.0	68.1	-0.1	31.8	53.5	65.6	+2.3	27.9	55.2	68.3	+2.9	66.9	67.4

Table 2: UAS of the various mono-source and multi-source transfer methods, on each UD target language (using UD language codes).
The first line reads as follows: for delexicalized transfer to Arabic, the worst, median and best sources yield UAS scores of 5.1, 43.2 and 56.9, and the average score over all 39 sources is 36.1, which the WALS-based method improves by 5.7 points.

Our experiments also show that the selection baseline method does not perform as well on Universal Dependencies (Nivre et al., 2016) as it did on the CoNLL 2006 Shared Task. Those differences can be explained in two ways. First, we experiment with cleaner treebanks and benefit from the availability of unified tagsets and annotation schemes. This is in contrast with previous experiments, which were using a tagset mapping as a preprocessing step, making the net effect of data selection more difficult to single out and evaluate precisely. Second, the data selection method was primarily intended for distantly related languages, whereas the UD corpus now offers a wide language diversity and often a few good sources for which data size reduction is only detrimental.

In general, our methods do not improve the best source but have a large effect on bad and average sources, often turning them into competitive sources. This is particularly true with PoSLM reordering, which improves the worst sources by 8.1 points and degrades the best ones by 2.6 points. By contrast, the WALS-based method is more conservative and offers lower but more reliable improvements, which in average proves successful.

Table 3 reports the average over some language families[12] of the UAS of the baseline, reordered and WALS-based mono-source models. It shows that accuracies of related sources are only marginally mod-

[12]While the considered ancient languages belong to some of those families, we chose to gather them into a separate category, since they rely on the same WALS completion heuristic, instead of their actual typological features.

Source language	Target language: Romance	Germanic	Slavic	Finno-Ugric	Semitic	Ancient
Romance	67.1‖65.6‖67.2	60.4‖60.4‖61.7	63.1‖63.5‖63.0	46.4‖50.8‖52.5	54.1‖52.1‖52.9	56.7‖56.5‖54.9
Germanic	61.2‖63.5‖65.8	65.9‖63.1‖65.8	61.3‖62.2‖63.2	57.2‖58.6‖58.5	41.2‖48.2‖49.8	54.5‖57.1‖56.7
Slavic	63.5‖61.7‖66.0	63.8‖60.5‖64.3	72.6‖68.4‖71.8	53.2‖57.0‖58.4	54.7‖53.6‖56.8	59.0‖59.2‖60.1
Finno-Ugric	46.3‖51.9‖52.3	57.1‖56.2‖57.6	53.8‖58.6‖56.9	64.1‖63.0‖64.2	30.0‖43.6‖41.5	50.8‖55.7‖56.1
Semitic	54.1‖54.2‖54.1	40.6‖48.2‖51.1	42.5‖54.6‖56.1	30.8‖41.2‖44.1	55.4‖55.6‖54.8	53.7‖55.9‖54.4
Ancient	56.1‖49.2‖55.9	56.7‖51.5‖56.1	60.9‖57.5‖60.6	52.2‖54.9‖56.0	51.1‖47.0‖50.6	62.7‖60.0‖62.6

Table 3: Delexicalized, PoSLM-based reordered and WALS-based UAS aggregated over language family pairs.
The first column reads as follows: the average UAS over all pairs of two Romance languages is 67.1 for mono-source delexicalized transfer; it is slightly improved (67.2) by the WALS-based method. Over all pairs of a Germanic source and a Romance target, the average mono-source UAS raises from 61.2 (delexicalized baseline) to 63.5 (PoSLM-based reordering) and 65.8 (WALS-based rules).

ified when source sentences are transformed according to WALS, which could be expected as related languages share most of their typological features. On the contrary, large gains are obtained for distantly related languages. Such languages are typically poor sources in direct delexicalized transfer due to systematic labeling errors that mostly concern few frequent word classes (in correlation with their typological features). We have found that such errors can often be corrected by transforming the source sentences. With those errors handled, the now competitive sources can in turn contribute with valuable knowledge in multi-source settings.

4.3 A Fine-grained Analysis

We have also investigated the improvements made over the baseline by our best method, the WALS-based rewriting rules, by analyzing the gain in accuracy separately for various PoS. It appears that, in most cases, improvements mostly concern PoS classes covered by the WALS features. For instance, the issue mentioned in § 2.2 for the English-French pair is almost solved with source reordering: 90.4% of the postposed ADJs are correctly predicted by the WALS-based method (34.5% in the baseline), without any detrimental impact on the preposed ones. The same holds for the Hebrew-Bulgarian textbook case, where the UAS $\left[\begin{smallmatrix}\text{NOUN}\\ \text{ADJ}\end{smallmatrix}\right]$ raises from 0.7% to 95.1%.

We observe similar behaviors across the board for all the classes targeted by transformations: transfer from Czech to Danish had UAS $\left[\begin{smallmatrix}\text{NOUN}\\ \text{preposed NOUN}\end{smallmatrix}\right]$ and UAS $\left[\begin{smallmatrix}\text{NOUN}\\ \text{postposed NOUN}\end{smallmatrix}\right]$ scores of 2.8% and 78.4%, with WALS-based preprocessing they are respectively 61.1% and 80.4%. In Finnish-Arabic, scores of 6.3% and 30.9% on ADJs and ADPs become 65.8% and 61.4%, etc. In whole, 21% of the considered language pairs present very large gains (over +50 points) for at least one frequent tag pair (over 30 dependency occurrences in test data).

Careful comparison of results for both PoSLM-based methods shows that reordering improves ADJs' attachment for instance, when data selection does not. This can be explained in two ways. First, if the source corpus contains a very limited number of preposed ADJs, even with perfect selection the ADJs in final data cannot be mostly preposed. Second, data selection mostly targets sequences very far from the target syntax: sentences that only disrespect a local preference of child position are less fluent, and consequently have a lower rank, than their hypothetical counterpart with switched positions; but they are not ungrammatical enough to be pushed into the 10% worst territory. On the other hand, the data transformation approach is not restricted to preexisting n-grams, and it directly confronts the given sequence with its counterpart to keep only the most fluent, thus acknowledging local preferences. These key differences are confirmed experimentally on English-French data: PoSLM-based reordering lowers the precedence rate of ADJs to NOUNs from 93% to 60%, while the rate varies by less than 1% in Søgaard (2011)'s approach, leaving the majority case adjectives still under-trained.

Finally, detailed analyses reveal that the PoSLM reordering approach has lower improvements than the WALS-based one on *easy reorderings* such as the nearly deterministic Hebrew-Bulgarian adjectives

(UAS $\left[\begin{smallmatrix}\text{NOUN}\\ \text{ADJ}\end{smallmatrix}\right]$ of 93.2% with WALS, versus 86.1% with the PoSLM). This suggests that the data-driven technique still wastes part of the available knowledge: indeed, the use of a probabilistic model to rank reorderings does not guarantee that any interesting reordering is in fact selected. Another advantage of the knowledge-rich approach is that the distribution of *local* word orderings is easier to control, since it explicitly regulates the balance between co-existing word orders. Indeed, when two structures are possible and fluent, the PoSLM-based method will always prefer the majority class. While the projectivity requirement generally softens this hard constraint by discarding many reordering candidates, the effect holds for instance on typologically close languages: during transfer from French ($\text{PR} = 28\%$ for ADJ-NOUN) to Italian ($\text{PR} = 32\%$), PoSLM-based reorderings harden the preference down to $\text{PR} = 16\%$, and end up under-training the ADJ-NOUN minority class.

In spite of this, the PoSLM reordering is still competitive, since it covers more diverse phenomena. For instance, when transferring from English to Tamil, the UAS $\left[\begin{smallmatrix}\text{VERB}\\ \text{AUX}\end{smallmatrix}\right]$ only raises from 31.4% to 35.0% with the WALS-based method, but achieves a nice 91.2% with the PoSLM. Such improvements are however less predictable and unexplained losses also occur, as for the UAS $\left[\begin{smallmatrix}\text{VERB}\\ \text{AUX}\end{smallmatrix}\right]$ in Hungarian-Tamil (98.5% for the baseline and WALS, 66.4% with PoSLM reordering).

These results suggest that both approaches have their upsides and downsides, which remain to be combined.

5 Related Work

The observation that cross-lingual transfer works better with typologically close or related languages has been already made by many. Indeed, several works have already pointed out that unified annotation schemes cannot compensate for syntactic divergences between source and target languages and that reducing these divergences was likely to improve the performance of transfer.

When several sources are available, a natural approach is to give more weight to the instances observed in related languages, where relatedness can be measured either based on linguistic description (Berg-Kirkpatrick and Klein, 2010) or empirically (Cohen et al., 2011).

Søgaard (2011) follows similar intuitions but binarizes the weights to apply instance selection. Thus, the delexicalized model is trained only on the source examples that are the most *relevant* for the target at hand, using PoSLM perplexity as a relevance metric. Note that this strategy can be applied both in mono-source and multi-source settings.

In Rosa and Zabokrtsky (2015)'s work, the syntactic similarity between languages is also based on the similarity between their PoS sequences. They show how the KL_{cpos^3} measure can be used to improve cross-lingual transfer either by selecting the best source language, or by weighting the source contribution to the output in a multi-source setting.

Both multi-source combination and data selection follow the same intuition that any source sentence or part of it can provide useful information on the syntax of the target language, even when the divergence between the source and the target is large. Indeed, a language is subject to many influences throughout its evolution and can borrow a phenomenon from a very distant language. This is for instance the case of Romanian, which belongs to the Romance family but has also strong Slavic influences.

As a result, both works aim at extracting useful knowledge even from poor sources, and our proposal can be viewed as an extension that pushes further this intuition, to a finer grain: Rosa and Zabokrtsky (2015) reward good source languages, Søgaard (2011) rewards target-relevant sentences, and our method rewards relevant local patterns, by performing a local reordering of target-irrelevant parse subtrees rather than ignoring the whole sentence. This has the effect of using the knowledge embedded in these subtrees as well as in the rest of the sentence more effectively. To see this, consider the case of transferring an English parser to a language in which no verb is labeled as auxiliary.[13] In this case, the method of (Søgaard, 2011) is likely to discard all the English sentences containing auxiliaries and the parser will hardly see, in training, sentences involving passive constructions or past participles; by contrast, methods based on data transformation would not remove the full sentence, but just the auxiliary – all the other dependencies, e.g. the by-agent, can still contribute to learning.

[13] In the UD treebank this is, for instance, the case for Greek, Latvian or Galician.

Thus, in comparison to previous works favoring close word orders at the cost of discarding some training examples or reducing source contribution, our method differs by improving cross-lingual transfer without knowledge loss.

In another line of work, Naseem et al. (2012) also distinguish the knowledge 'ADJs depend on NOUNs' from the ordering of both tokens, and use WALS to predict the latter. However, where our methods compensate for word order divergences at the data level, their work aims at abstracting the dependency prediction from word order, by designing a new parsing algorithm from scratch: their parser decomposes as a dependent selection component, shared among languages, and an ordering component that is specific to the target language. Even though it does not provide full order abstraction, our approach has the double advantage of wrapping any state-of-the-art parsing system, and allowing an extra degree of flexibility by manipulating the data, e.g. to handle PoS classes existing in only one language.

6 Conclusion

The contribution of this work is twofold. First, we have updated earlier results on delexicalized cross-lingual model transfer by reproducing them on the recent Universal Dependencies treebank. This collection of treebanks contains more languages than were previously available. Furthermore, the consistency of annotation schemes makes the analysis of results more reliable and enables to draw firmer conclusions. Second, based on a thorough analysis of the weaknesses of delexicalized transfer, we have proposed two strategies that aim to compensate for word sequence biases when transferring models across languages: a data-driven method using PoSLMs for reordering and a knowledge-based method exploiting heuristic rewrite rules extracted from WALS. The latter method proved to be the most effective of the two, with the additional benefit of being entirely resource-free and thus readily usable for the over thousand languages whose word order is specified in WALS. For the frequent PoS classes targeted by this method, we were able to obtain huge improvements, often 30 and up to 90 points.

A first natural continuation of this work will be to complete our repertoire of preprocessing rules with article insertions, PoS substitutions and patterns involving verbs, which were not considered so far. Another direction we would like to investigate is to contrast our techniques with annotation projection, which is another way to compensate for word order biases in cross-lingual transfer: by analyzing the pros and cons of each method we might find ways to combine them so that we can also use parallel data when available. We finally also aim at generalizing our WALS approach to other order-dependent tasks. Indeed, from a higher-level point of view, the aforementioned issues are not specific to dependency parsing, but occur theoretically with all data-driven NLP methods: however general it is, the linguistic knowledge is always only available as instantiated on a given word sequence and through the proxy of a particular data.

Acknowledgements

This work has been partly funded by the French *Direction générale de l'armement* and by the European Union's Horizon 2020 research and innovation programme under grant agreement No. 645452 (QT21). We thank the anonymous reviewers for their detailed and inspiring comments on the paper.

References

Lauriane Aufrant and Guillaume Wisniewski. 2016. PanParser: a Modular Implementation for Efficient Transition-Based Dependency Parsing. Technical report, LIMSI-CNRS, March.

Taylor Berg-Kirkpatrick and Dan Klein. 2010. Phylogenetic grammar induction. In *The 48th Annual Meeting of the Association for Computational Linguistics*, pages 1288–1297, Uppsala, Sweden.

Arianna Bisazza and Marcello Federico. 2016. A Survey of Word Reordering in Statistical Machine Translation: Computational Models and Language Phenomena. *Computational Linguistics*, 42(2):163–205.

Shay B. Cohen, Dipanjan Das, and Noah A. Smith. 2011. Unsupervised Structure Prediction with Non-Parallel Multilingual Guidance. In *Proceedings of EMNLP 2011, the Conference on Empirical Methods in Natural Language Processing*, pages 50–61, Edinburgh, Scotland, UK., July.

Michael Collins and Brian Roark. 2004. Incremental parsing with the perceptron algorithm. In *Proceedings of the 42Nd Annual Meeting on Association for Computational Linguistics*, ACL '04.

Markus Dreyer, Keith Hall, and Sanjeev Khudanpur. 2007. Comparing reordering constraints for smt using efficient bleu oracle computation. In *Proceedings of SSST, NAACL-HLT 2007 / AMTA Workshop on Syntax and Structure in Statistical Translation*, pages 103–110, Rochester, New York, April. Association for Computational Linguistics.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Max Planck Institute for Evolutionary Anthropology, Leipzig.

Dan Garrette and Jason Baldridge. 2013. Learning a part-of-speech tagger from two hours of annotation. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 138–147, Atlanta, Georgia.

Yoav Goldberg and Joakim Nivre. 2012. A Dynamic Oracle for Arc-Eager Dependency Parsing. In *Proceedings of COLING 2012, the International Conference on Computational Linguistics*, pages 959–976, Bombay, India.

Rebecca Hwa, Philip Resnik, A.Weinberg, C. Cabezas, and O. Kolak. 2005. Bootstrapping Parsers via Syntactic Projection accross Parallel Texts. *Natural language engineering*, 11:311–325.

Shankar Kumar and William Byrne. 2005. Local phrase reordering models for statistical machine translation. In *Proceedings of Human Language Technology Conference and Conference on Empirical Methods in Natural Language Processing*, pages 161–168, Vancouver, British Columbia, Canada.

Ophélie Lacroix, Lauriane Aufrant, Guillaume Wisniewski, and François Yvon. 2016. Frustratingly Easy Cross-Lingual Transfer for Transition-Based Dependency Parsing. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1058–1063.

Adam Lopez. 2009. Translation as weighted deduction. In *Proceedings of the 12th Conference of the European Chapter of the Association for Computational Linguistics*, EACL '09, pages 532–540, Stroudsburg, PA, USA. Association for Computational Linguistics.

Ryan McDonald, Joakim Nivre, Yvonne Quirmbach-Brundage, Yoav Goldberg, Dipanjan Das, Kuzman Ganchev, Keith Hall, Slav Petrov, Hao Zhang, Oscar Täckström, Claudia Bedini, Núria Bertomeu Castelló, and Jungmee Lee. 2013. Universal Dependency Annotation for Multilingual Parsing. In *Proceedings of ACL 2013, the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 92–97, Sofia, Bulgaria, August.

Tahira Naseem, Regina Barzilay, and Amir Globerson. 2012. Selective sharing for multilingual dependency parsing. In *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers - Volume 1*, ACL '12, pages 629–637.

Joakim Nivre, Marie-Catherine de Marneffe, Filip Ginter, Yoav Goldberg, Jan Hajic, Christopher D. Manning, Ryan McDonald, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Reut Tsarfaty, and Daniel Zeman. 2016. Universal dependencies v1: A multilingual treebank collection. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC 2016)*. European Language Resources Association (ELRA).

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

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In Nicoletta Calzolari (Conference Chair), Khalid Choukri, Thierry Declerck, Mehmet Uğur Doğan, Bente Maegaard, Joseph Mariani, Jan Odijk, and Stelios Piperidis, editors, *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey, may. European Language Resources Association (ELRA).

Rudolf Rosa and Zdenek Zabokrtsky. 2015. KL_{cpos^3} - a Language Similarity Measure for Delexicalized Parser Transfer. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 243–249, Beijing, China. Association for Computational Linguistics.

Anders Søgaard. 2011. Data point selection for cross-language adaptation of dependency parsers. In *Proceedings of ACL 2011, the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 682–686, Portland, Oregon, USA, June.

Jörg Tiedemann, Željko Agić, and Joakim Nivre. 2014. Treebank translation for cross-lingual parser induction. In *Proceedings of the Eighteenth Conference on Computational Natural Language Learning (CoNLL 2014)*, pages 130–140, Ann Arbor, Michigan.

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 Chapter of the Association for Computational Linguistics on Language technologies*, NAACL '01, pages 1–8, Stroudsburg, PA, USA.

Daniel Zeman and Philip Resnik. 2008. Cross-Language Parser Adaptation between Related Languages. In *Proceedings of the IJCNLP-08 Workshop on NLP for Less Privileged Languages*, pages 35–42, Hyderabad, India, January. Asian Federation of Natural Language Processing.

Yue Zhang and Joakim Nivre. 2011. Transition-based dependency parsing with rich non-local features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 188–193, Portland, Oregon, USA.