Cross-lingual Transfer for Unsupervised Dependency Parsing Without Parallel Data

Long Duong,[12] **Trevor Cohn,**[1] **Steven Bird,**[1] and **Paul Cook**[3]
[1]Department of Computing and Information Systems, University of Melbourne
[2]National ICT Australia, Victoria Research Laboratory
[3]Faculty of Computer Science, University of New Brunswick
lduong@student.unimelb.edu.au {t.cohn,sbird}@unimelb.edu.au paul.cook@unb.ca

Abstract

Cross-lingual transfer has been shown to produce good results for dependency parsing of resource-poor languages. Although this avoids the need for a target language treebank, most approaches have still used large parallel corpora. However, parallel data is scarce for low-resource languages, and we report a new method that does not need parallel data. Our method learns syntactic word embeddings that generalise over the syntactic contexts of a bilingual vocabulary, and incorporates these into a neural network parser. We show empirical improvements over a baseline delexicalised parser on both the CoNLL and Universal Dependency Treebank datasets. We analyse the importance of the source languages, and show that combining multiple source-languages leads to a substantial improvement.

1 Introduction

Dependency parsing is a crucial component of many natural language processing (NLP) systems for tasks such as relation extraction (Bunescu and Mooney, 2005), statistical machine translation (Xu et al., 2009), text classification (Özgür and Güngör, 2010), and question answering (Cui et al., 2005). Supervised approaches to dependency parsing have been very successful for many resource-rich languages, where relatively large treebanks are available (McDonald et al., 2005a). However, for many languages, annotated treebanks are not available, and are very costly to create (Böhmová et al., 2001). This motivates the development of unsupervised approaches that can make use of unannotated, monolingual data. However, purely unsupervised approaches have relatively low accuracy (Klein and Manning, 2004; Gelling et al., 2012).

Most recent work on unsupervised dependency parsing for low-resource languages has used the idea of delexicalized parsing and cross-lingual transfer (Zeman et al., 2008; Søgaard, 2011; McDonald et al., 2011; Ma and Xia, 2014). In this setting, a delexicalized parser is trained on a resource-rich *source* language, and is then applied directly to a resource-poor *target* language. The only requirement here is that the source and target languages are POS tagged must use the same tagset. This assumption is pertinent for resource-poor languages since it is relatively quick to manually POS tag the data. Moreover, there are many reports of high accuracy POS tagging for resource-poor languages (Duong et al., 2014; Garrette et al., 2013; Duong et al., 2013b). The cross-lingual delexicalized approach has been shown to significantly outperform unsupervised approaches (McDonald et al., 2011; Ma and Xia, 2014).

Parallel data can be used to boost the performance of a cross-lingual parser (McDonald et al., 2011; Ma and Xia, 2014). However, parallel data may be hard to acquire for truly resource-poor languages.[1] Accordingly, we propose a method to improve the performance of a cross-lingual delexicalized parser using only monolingual data.

Our approach is based on augmenting the delexicalized parser using syntactic word embeddings. Words from both source and target language are mapped to a shared low-dimensional space based on their syntactic context, without recourse to parallel data. While prior work has struggled to efficiently incorporate word embedding information into the parsing model (Bansal et al., 2014; Andreas and Klein, 2014; Chen et al., 2014), we present a method for doing so using a neural net-

[1] Note that most research in this area (as do we) evaluates on simulated low-resource languages, through selective use of data in high-resource languages. Consequently parallel data is plentiful, however this is often not the case in the real setting, e.g., for Tagalog, where only scant parallel data exists (e.g., dictionaries, Wikipedia and the Bible).

Proceedings of the 19th Conference on Computational Language Learning, pages 113–122,
Beijing, China, July 30-31, 2015.

work parser. We train our parser using a two stage process: first learning cross-lingual syntactic word embeddings, then learning the other parameters of the parsing model using a source language treebank. When applied to the target language, we show consistent gains across all studied languages.

This work is a stepping stone towards the more ambitious goal of a universal parser that can efficiently parse many languages with little modification. This aspiration is supported by the recent release of the Universal Dependency Treebank (Nivre et al., 2015) which has consensus dependency relation types and POS annotation for many languages.

When multiple source languages are available, we can attempt to boost performance by choosing the best source language, or combining information from several source languages. To the best of our knowledge, no prior work has proposed a means for selecting the best source language given a target language. To address this, we introduce two metrics which outperform the baseline of always picking English as the source language. We also propose a method for combining all available source languages which leads to substantial improvement.

The rest of this paper is organized as follows: Section 2 reviews prior work on unsupervised cross-lingual dependency parsing. Section 3 presents the methods for improving the delexicalized parser using syntactic word embeddings. Section 4 describes experiments on the CoNLL dataset and Universal Dependency Treebank. Section 5 presents methods for selecting the best source language given a target language.

2 Unsupervised Cross-lingual Dependency Parsing

There are two main approaches for building dependency parsers for resource-poor languages without using target-language treebanks: delexicalized parsing and projection (Hwa et al., 2005; Ma and Xia, 2014; Täckström et al., 2013; McDonald et al., 2011).

The delexicalized approach was proposed by Zeman et al. (2008). They built a delexicalized parser from a treebank in a resource-rich source language. This parser can be trained using any standard supervised approach, but without including any lexical features, then applied directly to parse sentences from the resource-poor language. Delexicalized parsing relies on the fact that parts-of-speech are highly informative of dependency relations. For example, an English lexicalized discriminative arc-factored dependency parser achieved 84.1% accuracy, whereas a delexicalized version achieved 78.9% (McDonald et al., 2005b; Täckström et al., 2013). Zeman et al. (2008) build a parser for Swedish using Danish, two closely-related languages. Søgaard (2011) adapt this method for less similar languages by choosing sentences from the source language that are similar to the target language. Täckström et al. (2012) additionally use cross-lingual word clustering as a feature for their delexicalized parser. Also related is the work by Naseem et al. (2012) and Täckström et al. (2013) who incorporated linguistic features from the World Atlas of Language Structures (WALS; Dryer and Haspelmath (2013)) for joint modelling of multi-lingual syntax.

In contrast, projection approaches use parallel data to project source language dependency relations to the target language (Hwa et al., 2005). Given a source-language parse tree along with word alignments, they generate the target-language parse tree by projection. However, their approach relies on many heuristics which would be difficult to adapt to other languages. McDonald et al. (2011) exploit both delexicalized parsing and parallel data, using an English delexicalized parser as the seed parser for the target languages, and updating it according to word alignments. The model encourages the target-language parse tree to look similar to the source-language parse tree with respect to the head-modifier relation. Ma and Xia (2014) use parallel data to transfer source language parser constraints to the target side via word alignments. For the null alignment, they used a delexicalized parser instead of the source language lexicalized parser.

In summary, existing work generally starts with a delexicalized parser, and uses parallel data typological information to improve it. In contrast, we want to improve the delexicalized parser, but without using parallel data or any explicit linguistic resources.

3 Improving Delexicalized Parsing

We propose a novel method to improve the performance of a delexicalized cross-lingual parser without recourse to parallel data. Our method uses no additional resources and is designed to complement other methods. The approach is based on syntactic word embeddings where a word is represented as a low-dimensional vector in syntactic space. The idea is simple: we want to relexicalize the delexicalized parser using word embeddings, where source and target language lexical items are represented in the same space.

Word embeddings typically capture both syntactic and semantic information. However, we hypothesize (and later show empirically) that for dependency parsing, word embeddings need to better reflect syntax. In the next subsection, we review some cross-lingual word embedding methods and propose our syntactic word embeddings. Section 4 empirically compares these word embeddings when incorporated into a dependency parser.

3.1 Cross-lingual word embeddings

We review methods that can represent words in both source and target languages in a low-dimensional space. There are many benefits of using a low-dimensional space. Instead of the traditional "one-hot" representation with the number of dimensions equal to vocabulary size, words are represented using much fewer dimensions. This confers the benefit of generalising over the vocabulary to alleviate issues of data sparsity, through learning representations encoding lexical relations such as synonymy.

Several approaches have sought to learn cross-lingual word embeddings from parallel data (Hermann and Blunsom, 2014a; Hermann and Blunsom, 2014b; Xiao and Guo, 2014; Zou et al., 2013; Täckström et al., 2012). Hermann and Blunsom (2014a) induced a cross-lingual word representation based on the idea that representations for parallel sentences should be close together. They constructed a sentence level representation as a bag-of-words summing over word-level representations, and then optimized a hinge loss function to match a latent representation of both sides of a parallel sentence pair. While this might seem well suited to our needs as a word representation in cross-lingual parsing, it may lead to overly semantic embeddings, which are important for translation, but less useful for parsing. For example, "*economic*" and "*economical*" will have a similar representation despite having different syntactic features.

Also related is (Täckström et al., 2012) who build cross-lingual word representations using a variant of the Brown clusterer (Brown et al., 1992) applied to parallel data. Bansal et al. (2014) and Turian et al. (2010) showed that for monolingual dependency parsing, the simple Brown clustering based algorithm outperformed many word embedding techniques. In this paper we compare our approach to forming cross-lingual word embeddings with those of both Hermann and Blunsom (2014a) and Täckström et al. (2012).

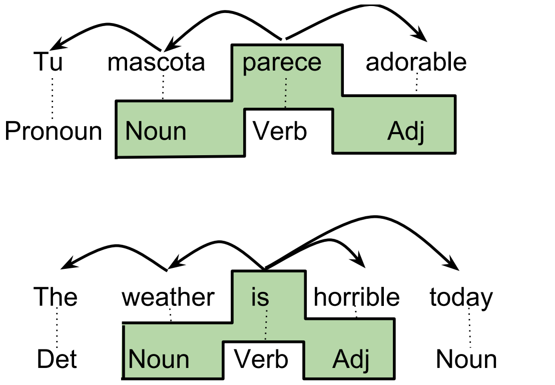

Figure 1: Examples of the syntactic word embeddings for Spanish and English. In each case, the highlighted tags are predicted by the highlighted word. The Spanish sentence means "*your pet looks lovely*".

3.2 Syntactic Word Embedding

We now propose a novel approach for learning cross-lingual word embeddings that is more heavily skewed towards syntax. Word embedding methods typically exploit word co-occurrences, building on traditional techniques for distributional similarity, e.g., the co-occurrences of words in a context window about a central word. Bansal et al. (2014) suggested that for dependency parsing, word embeddings be trained over dependency relations, instead of adjacent tokens, such that embeddings capture head and modifier relations. They showed that this strategy performed much better than surface embeddings for monolingual dependency parsing. However, their method is not applicable to our low resource setting, as it requires a parse tree for training. Instead we consider a simpler representation, namely part-of-speech contexts. This requires only POS tagging, rather than full parsing, while providing syntactic information linking words to their POS context, which we expect to be informative for characterising dependency relations.

Algorithm 1 Syntactic word embedding

```
1: Match the source and target tagsets to the Universal Tagset.
2: Extract word n-gram sequences for both the source and target language.
3: For each n-gram, keep the middle word, and replace the other words by their POS.
4: Train a skip-gram word embedding model on the resulting list of word and POS sequences from both the source and target language
```

We assume the same POS tagset is used for both the source and target language,[2] and learn word embeddings for each word type in both languages into the same syntactic space of nearby POS contexts. In particular, we develop a predictive model of the tags to the left and right of a word, as illustrated in Figure 1 and outlined in Algorithm 1. Figure 1 illustrates two training contexts extracted from our English source and Spanish target language, where the highlighted fragments reflect the tags being predicted around each focus word. Note that for this example, the POS contexts for the English and Spanish verbs are identical, and therefore the model would learn similar word embeddings for these terms, and bias the parser to generate similar dependency structures for both terms.

There are several motivations for our approach: (1) POS tags are too coarse-grained for accurate parsing, but with access to local context they can be made more informative; (2) leaving out the middle tag avoids duplication because this is already known to the parser; (3) dependency edges are often local, as shown in Figure 1, i.e., there are dependency relations between most words and their immediate neighbours. Consequently, training our embeddings to predict adjacent tags is likely to learn similar information to training over dependency edges.[3] Bansal et al. (2014) studied the effect of word embeddings on dependency parsing, and found that larger embedding windows captured more semantic information, while smaller windows better reflected syntax. Therefore we choose a small ±1 word window in our experiments. We also experimented with bigger windows ($\pm 2, \pm 3$) but observed performance degradation in these cases, supporting the argument above.

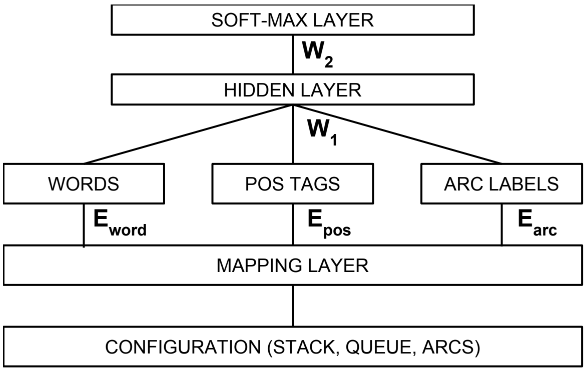

Figure 2: Neural Network Parser Architecture from Chen and Manning (2014)

Step 4 of Algorithm 1 finds the word embeddings as a side-effect of training a neural language model. We use the skip-gram model (Mikolov et al., 2013), trained to predict context tags for each word. The model is formulated as a simple bilinear logistic classifier

$$P(t_c|w) = \frac{\exp(\mathbf{u}_{t_c}^\top \mathbf{v}_w)}{\sum_{z=1}^{T} \exp(\mathbf{u}_z^\top \mathbf{v}_w)} \qquad (1)$$

where t_c is the context tag around the current word w, $\mathbf{U} \in \mathbb{R}^{T \times D}$ is the tag embedding matrix, $\mathbf{V} \in \mathbb{R}^{V \times D}$ is the word embedding matrix, with T the number of tags, V is the total number of word types over both languages and D the capacity of the embeddings. Given a training set of word and POS contexts, $(t_i^L, w_i, t_i^R)_{i=1}^N$,[4] we maximize the log-likelihood $\sum_{i=1}^N \log P(t_i^L|w_i) + \log P(t_i^R|w_i)$ with respect to $\mathbf{U}$ and $\mathbf{V}$ using stochastic gradient descent. The learned $\mathbf{V}$ matrix of word embeddings is later used in parser training (the source word embeddings) and inference (the target word embeddings).

3.3 Parsing Algorithm

In this Section, we show how to incorporate the syntactic word embeddings into a parsing model. Our parsing model is built based on the work of Chen and Manning (2014). They built a transition-based dependency parser using a neural-network. The neural network classifier will decide which transition is applied for each configuration.

[2] Later we consider multiple source languages, but for now assume a single source language.

[3] For the 16 languages in the CoNLL-X and CoNLL-07 datasets we observed that approx. 50% of dependency relations span a distance of one word and 20% span two words. Thus our POS context of a ±1 word window captures the majority of dependency relations.

[4] Note that w here can be a word type in either the source or target language, such that both embeddings will be learned for all word types in both languages.

The architecture of the parser is illustrated in Figure 2, where each layer is fully connected to the layer above.

For each configuration, the selected list of words, POS tags and labels from the Stack, Queue and Arcs are extracted. Each word, POS or label is mapped to a low-dimension vector representation (embedding) through the Mapping Layer. This layer simply concatenates the embeddings which are then fed into a two-layer neural network classifier to predict the next parsing action. The set of parameters for the neural network classifier is $E_{word}, E_{pos}, E_{labels}$ for the mapping layer, W_1 for the hidden layer and W_2 for the soft-max output layer. We incorporate the syntactic word embeddings into the neural network model by setting E_{word} to the syntactic word embeddings, which remain fixed during training so as to retain the cross-lingual mapping.[5]

3.4 Model Summary

To apply the parser to a resource-poor target language, we start by building syntactic word embeddings between source and target languages as shown in algorithm 1. Next we incorporate syntactic word embeddings using the algorithm proposed in Section 3.3. The third step is to substitute source- with target-language syntactic word embeddings. Finally, we parse the target language using this substituted model. In this way, the model will recognize lexical items for the target language.

4 Experiments

We test our method of incorporating syntactic word embeddings into a neural network parser, for both the existing CoNLL dataset (Buchholz and Marsi, 2006; Nivre et al., 2007) and the newly-released Universal Dependency Treebank (Nivre et al., 2015). We employed the Unlabeled Attachment Score (UAS) without punctuation for comparison with prior work on the CoNLL dataset. Where possible we also report Labeled Attachment Score (LAS) without punctuation. We use English as the source language for this experiment.

4.1 Experiments on CoNLL Data

In this section we report experiments involving the CoNLL-X and CoNLL-07 datasets. Running on this dataset makes our model comparable with prior work. For languages included in both datasets, we use the newer one only. Crucially, for the delexicalized parser we map language-specific tags to the universal tagset (Petrov et al., 2012). The syntactic word embeddings are trained using POS information from the CoNLL data.

There are two baselines for our experiment. The first one is the unsupervised dependency parser of Klein and Manning (2004), the second one is the delexicalized parser of Täckström et al. (2012). We also compare our syntactic word embedding with the cross-lingual word embeddings of Hermann and Blunsom (2014a). These word embeddings are induced by running each language pair using Europarl (Koehn, 2005). We incorporated Hermann and Blunsom (2014a)'s cross-lingual word embeddings into the parsing model in the same way as for the syntactic word embeddings. Table 1 shows the UAS for 8 languages for several models. The first observation is that the direct transfer delexicalized parser out-performed the unsupervised approach. This is consistent with many prior studies. Our implementation of the direct transfer model performed on par with Täckström et al. (2012) on average. Table 1 also shows that using HB embeddings improve the performance over the Direct Transfer model. Our model using syntactic word embedding consistently out-performed the Direct Transfer model and HB embedding across all 8 languages. On average, it is 1.5% and 1.3% better. [6] The improvement varies across languages compared with HB embedding, and falls in the range of 0.3 to 2.6%. This confirms our initial hypothesis that we need word embeddings that capture syntactic instead of semantic information.

It is not strictly fair to compare our method with prior approaches to unsupervised dependency parsing, since they have different resource requirement, i.e. parallel data or typological resources. Compared with the baseline of the direct transfer model, our approach delivered a 1.5% mean performance gain, whereas Täckström et al. (2012) and McDonald et al. (2011) report approximately 3% gain, Ma and Xia (2014) and Naseem et al. (2012) report an approximately 6% gain. As we

[5] This is a consequence of only training the parser on the source language. If we were to update embeddings during parser training this would mean they no longer align with the target language embeddings.

[6] All performance comparisons in this paper are absolute.

	da	de	el	es	it	nl	pt	sv	Avg
Unsupervised	33.4	18.0	39.9	28.5	43.1	38.5	20.1	44.0	33.2
Täckström et al. (2012) DT	36.7	48.9	59.5	60.2	64.4	52.8	66.8	55.4	55.6
Our Direct Transfer	44.1	44.9	63.3	52.2	57.7	59.7	67.5	55.4	55.6
Our Model + HB embedding	45.0	44.5	63.8	52.2	56.7	59.8	68.7	55.6	55.8
Our Model + Syntactic embedding	45.9	45.9	64.1	52.9	59.1	61.1	69.5	58.1	57.1

Table 1: Comparative results on the CoNLL corpora showing UAS for several parsers: unsupervised induction Klein and Manning (2004), Direct Transfer (DT) delexicalized parser of Täckström et al. (2012), our implementation of Direct Transfer and our neural network parsing model using cross-lingual embeddings Hermann and Blunsom (2014a) (HB) and our proposed syntactic embeddings.

	cs	de	en	es	fi	fr	ga	hu	it	sv
Train	1173.3	269.6	204.6	382.4	162.7	354.7	16.7	20.8	194.1	66.6
Dev	159.3	12.4	25.1	41.7	9.2	38.9	3.2	3.0	10.5	9.8
Test	173.9	16.6	25.1	8.5	9.1	7.1	3.8	2.7	10.2	20.4
Total	1506.5	298.6	254.8	432.6	181	400.7	23.7	26.5	214.8	96.8

Table 2: Number of tokens (× 1000) for each language in the Universal Dependency Treebank.

have stated above, our approach is complementary to the approaches used in these other systems. For example, we could incorporate the cross-lingual word clustering feature (Täckström et al., 2012) or WALS features (Naseem et al., 2012) into our model, or use our improved delexicalized parser as the reference model for Ma and Xia (2014), which we expect would lead to better results yet.

4.2 Experiments with Universal Dependency Treebank

We also experimented with the Universal Dependency Treebank V1.0, which has many desirable properties for our system, e.g. dependency types and coarse POS are the same across languages. This removes the need for mapping the source and target language tagsets to a common tagset, as was done for the CoNLL data. Secondly, instead of only reporting UAS we can report LAS, which is impossible on CoNLL dataset where the dependency edge labels differed among languages.

Table 2 shows the size in thousands of tokens for each language in the treebank. The first thing to observe is that some languages have abundant amount of data such as Czech (cs), French (fr) and Spanish (es). However, there are languages with modest size i.e. Hungarian (hu) and Irish (ga).

We ran our model with and without syntactic word embeddings for all languages with English as the source language. The results are shown in Table 3. The first observation is that our model using syntactic word embeddings out-performed direct transfer for all the languages on both UAS and LAS. We observed an average improvement of 3.6% (UAS) and 3.1% (LAS). This consistent improvement shows the robustness of our method of incorporating syntactic word embedding to the model. The second observation is that the gap between UAS and LAS is as big as 13% on average for both models. This reflects the increase difficulty of labelling the edges, with unlabelled edge prediction involving only a 3-way classification[7] while labelled edge prediction involves an 81-way classification.[8] Narrowing the gap between UAS and LAS for resource-poor languages is an important research area for future work.

5 Different Source Languages

In the previous sections, we used English as the source language. However, English might not be the best choice. For the delexicalized parser, it is crucial that the source and target languages have similar syntactic structures. Therefore a different choice of source language might substantially change the performance, as observed in prior studies (Täckström et al., 2013; Duong et al., 2013a; McDonald et al., 2011).

[7] Since there are only 3 transitions: SHIFT, LEFT-ARC, RIGHT-ARC.

[8] Since the Universal Dependency Treebank has 40 universal relations, each relation is attached to LEFT-ARC or RIGHT-ARC. The number 81 comes from 1 (SHIFT) + 40 (LEFT-ARC) + 40 (RIGHT-ARC).

	cs	de	es	fi	fr	ga	hu	it	sv	UAS	LAS
Direct Transfer	47.2	57.9	64.7	44.9	64.8	49.1	47.8	64.9	55.5	55.2	42.7
Our Model + Syntactic embedding	50.2	60.9	67.9	51.4	66.0	51.6	52.3	69.2	59.6	58.8	45.8

Table 3: Results comparing a direct transfer parser and our model with syntactic word embeddings. Evaluating UAS over the Universal Dependency Treebank. (We observed a similar pattern for LAS.) The rightmost UAS and LAS columns shows the average scores for the respective metric across 9 languages.

		Target Language											
		cs	de	en	es	fi	fr	ga	hu	it	sv	UAS	LAS
Source Language	cs	76.8	**65.9**	60.8	70.0	**53.7**	66.8	**59.0**	55.2	70.7	56.8	62.1	38.7
	de	60.0	78.2	61.7	63.1	52.4	60.6	49.8	**56.7**	64.0	59.5	58.6	45.5
	en	50.2	60.9	81.0	67.9	51.4	66.0	51.6	52.3	69.2	**59.6**	58.8	45.8
	es	**60.5**	58.5	60.4	80.9	45.7	**73.3**	53.8	46.9	**77.4**	55.3	59.1	46.2
	fi	49.0	41.8	44.5	33.6	71.5	35.2	24.4	44.6	31.7	43.1	38.7	25.5
	fr	54.2	55.7	**63.2**	**74.8**	43.6	79.2	54.7	44.3	76.2	54.8	57.9	46.3
	ga	32.8	35.3	39.8	56.3	23.5	52.6	72.3	26.0	58.3	32.6	39.7	26.7
	hu	42.3	53.4	45.4	43.8	53.3	42.1	29.2	72.1	41.2	42.5	43.7	22.7
	it	57.6	53.4	53.2	72.1	42.7	71.4	54.7	42.2	85.9	54.2	55.7	45.0
	sv	49.1	59.2	54.9	59.8	47.9	55.7	48.5	52.7	62.2	78.4	54.4	41.2

Table 4: UAS for each language pair in the Universal Dependency Treebank using our best model. The UAS/LAS column show the average UAS/LAS for all target languages, excluding the source language. The best UAS for each target language is shown in bold.

In this section we assume that we have multiple source languages. To see how the performance changes when using a different source language, we run our best model (i.e., using syntactic embeddings) for each language pair in the Universal Dependency Treebank. Table 4 shows the UAS for each language pair, and the average across all target languages for each source language. We also considered LAS, but observed similar trends, and therefore only report the average LAS for each source language. Observe that English is rarely the best source language; Czech and French give a higher average UAS and LAS, respectively. Interestingly, while Czech gives high UAS on average, it performs relatively poorly in terms of LAS.

One might expect that the relative performance from using different source languages is affected by the source corpus size, which varies greatly. We tested this question by limiting the source corpora 66K sentences (and excluded the very small *ga* and *hu* datasets), which resulted in a slight reduction in scores but overall a near identical pattern of results to the use of the full sized source corpora reported in Table 4. Only in one instance did the best source language change (for target *fi* with source *de* not *cs*), and the average rankings by UAS and LAS remained unchanged.

The ten languages considered belong to five families: *Romance* (French, Spanish, Italian), *Germanic* (German, English, Swedish), *Slavic* (Czech), *Uralic* (Hungarian, Finnish), and *Celtic* (Irish). At first glance it seems that language pairs in the same family tend to perform well. For example, the best source language for both French and Italian is Spanish, while the best source language for Spanish is French. However, this doesn't hold true for many target languages. For example, the best source language for both Finnish and German is Czech. It appears that the best choice of an appropriate source language is not predictable from language family information.

We therefore propose two methods to predict the best source language for a given target language. In devising these methods we assume that for a given resource-poor target language we do not have access to any parsed data, as this is expensive to construct. The first method is based on the Jensen-Shannon divergence between the distributions of POS n-grams ($1 < n < 6$) in a pair of languages. The second method converts each language into a vector of binary features based on word-order information from WALS, the World

	cs	de	en	es	fi	fr	ga	hu	it	sv	UAS	LAS
English	50.2	60.9	—	67.9	51.4	66.0	51.6	52.3	69.2	59.6	58.8	45.8
WALS	50.2	59.2	44.5	72.1	51.4	73.3	53.8	44.6	77.4	59.6	60.2	47.1
POS	49.1	58.5	53.2	74.8	53.7	73.3	53.8	56.7	76.2	56.8	61.4	47.7
Oracle	60.5	65.9	63.2	74.8	53.7	73.3	59.0	56.7	77.4	59.6	64.5	50.8
Combined	61.1	67.5	64.4	75.1	54.2	72.8	58.7	57.9	76.7	60.5	64.9	52.0

Table 5: UAS for target languages where the source language is selected in different ways. English uses English as the source language. WALS and POS choose the best source language using the WALS or POS ngrams based methods, respectively. Oracle always uses the best source language. Combined is the model that combines information from all available sources language. The UAS/LAS columns show the UAS/LAS average performance across 9 languages (English is excluded).

Atlas of Language Structures (Dryer and Haspelmath, 2013). These features include the relative order of adjective and noun, etc, and we compute the cosine similarity between the vectors for a pair of languages.

As an alternative to selecting a single source language, we further propose a method to combine information from all available source languages to build a parser for a target language. To do so we first train the syntactic word embeddings on all the languages. After this step, lexical items from all source languages and the target language will be in the same space. We train our parser with syntactic word embeddings on the combined corpus of all source languages. This parser is then applied to the target language directly. The intuition here is that training on multiple source languages limits over-fitting to the source language, and learns the "universal" structure of languages.

Table 5 shows the performance of each target language with the source language given by the model (in the case of models that select a single source language). Always choosing English as the source language performs worst. Using WALS features out-performs English on 7 out of 9 languages. Using POS ngrams out-performs the WALS feature model on average for both UAS and LAS, although the improvement is small. The combined model, which combines information from all available source languages, out-performs choosing a single source language. Moreover, this model performs even better than the oracle model, which always chooses the single best source language, especially for LAS. Compared with the baseline of always choosing English, our combined model gives an improvement about 6% for both UAS and LAS.

6 Conclusions

Most prior work on cross-lingual transfer dependency parsing has relied on large parallel corpora. However, parallel data is scarce for resource-poor languages. In the first part of this paper we investigated building a dependency parser for a resource-poor language without parallel data. We improved the performance of a delexicalized parser using syntactic word embeddings using a neural network parser. We showed that syntactic word embeddings are better at capturing syntactic information, and particularly suitable for dependency parsing. In contrast to the state-of-the-art for unsupervised cross-lingual dependency parsing, our method does not rely on parallel data. Although the state-of-the-art achieves bigger gains over the baseline than our method, our approach could be more-widely applied to resource-poor languages because of its lower resource requirements. Moreover, we have described how our method could be used to complement previous approaches.

The second part of this paper studied ways of improving performance when multiple source languages are available. We proposed two methods to select a single source language that both lead to improvements over always choosing English as the source language. We then showed that we can further improve performance by combining information from all the source languages. In summary, without any parallel data, we managed to improve the direct transfer delexicalized parser by about 10% for both UAS and LAS on average, for 9 languages in the Universal Dependency Treebank.

In this paper we focused only on word embeddings, however, in future work we could also build the POS embeddings and the arc-label embeddings across languages. This could help our system to move more freely across languages, facilitating not only the development of NLP for resource-poor languages, but also cross-language comparisons.

Acknowledgments

This work was supported by the University of Melbourne and National ICT Australia (NICTA). Dr Cohn is the recipient of an Australian Research Council Future Fellowship (project number FT130101105).

References

Jacob Andreas and Dan Klein. 2014. How much do word embeddings encode about syntax? In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 822–827, Baltimore, Maryland.

Mohit Bansal, Kevin Gimpel, and Karen Livescu. 2014. Tailoring continuous word representations for dependency parsing. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 809–815.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2001. The Prague Dependency Treebank: A Three-Level Annotation Scenario. In Anne Abeillé, editor, *Treebanks: Building and Using Syntactically Annotated Corpora*, pages 103–127.

Peter F. Brown, Peter V. deSouza, Robert L. Mercer, Vincent J. Della Pietra, and Jenifer C. Lai. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18:467–479.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*.

Razvan C. Bunescu and Raymond J. Mooney. 2005. A shortest path dependency kernel for relation extraction. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 724–731.

Danqi Chen and Christopher Manning. 2014. A fast and accurate dependency parser using neural networks. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 740–750, Doha, Qatar.

Wenliang Chen, Yue Zhang, and Min Zhang. 2014. Feature embedding for dependency parsing. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 816–826, Dublin, Ireland.

Hang Cui, Renxu Sun, Keya Li, Min-Yen Kan, and Tat-Seng Chua. 2005. Question answering passage retrieval using dependency relations. In *Proceedings of the 28th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '05, pages 400–407, New York, NY, USA.

Matthew S. Dryer and Martin Haspelmath, editors. 2013. *WALS Online*. Leipzig.

Long Duong, Paul Cook, Steven Bird, and Pavel Pecina. 2013a. Increasing the quality and quantity of source language data for Unsupervised Cross-Lingual POS tagging. In *Proceedings of the Sixth International Joint Conference on Natural Language Processing*, pages 1243–1249, Nagoya, Japan.

Long Duong, Paul Cook, Steven Bird, and Pavel Pecina. 2013b. Simpler unsupervised POS tagging with bilingual projections. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 634–639, Sofia, Bulgaria.

Long Duong, Trevor Cohn, Karin Verspoor, Steven Bird, and Paul Cook. 2014. What can we get from 1000 tokens? a case study of multilingual pos tagging for resource-poor languages. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 886–897, Doha, Qatar.

Dan Garrette, Jason Mielens, and Jason Baldridge. 2013. Real-world semi-supervised learning of postaggers for low-resource languages. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (ACL-2013)*, pages 583–592, Sofia, Bulgaria.

Douwe Gelling, Trevor Cohn, Phil Blunsom, and Joo Graa. 2012. The pascal challenge on grammar induction.

Karl Moritz Hermann and Phil Blunsom. 2014a. Multilingual Distributed Representations without Word Alignment. In *Proceedings of ICLR*.

Karl Moritz Hermann and Phil Blunsom. 2014b. Multilingual models for compositional distributed semantics. *CoRR*, abs/1404.4641.

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

Dan Klein and Christopher Manning. 2004. Corpus-based induction of syntactic structure: Models of dependency and constituency. In *Proceedings of the 42nd Annual Meeting on Association for Computational Linguistics*, ACL '04.

Philipp Koehn. 2005. Europarl: A Parallel Corpus for Statistical Machine Translation. In *Proceedings of the Tenth Machine Translation Summit (MT Summit X)*, pages 79–86, Phuket, Thailand.

Xuezhe Ma and Fei Xia. 2014. Unsupervised dependency parsing with transferring distribution via parallel guidance and entropy regularization. In *Proceedings of the 52nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1337–1348.

Ryan McDonald, Koby Crammer, and Fernando Pereira. 2005a. Online large-margin training of dependency parsers. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, ACL '05, pages 91–98.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajič. 2005b. Non-projective dependency parsing using spanning tree algorithms. In *Proceedings of the Conference on Human Language Technology and Empirical Methods in Natural Language Processing*, HLT '05, pages 523–530.

Ryan McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '11, pages 62–72.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In C.j.c. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.q. Weinberger, editors, *Advances in Neural Information Processing Systems 26*, pages 3111–3119.

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, Johan Hall, Sandra Kübler, Ryan McDonald, Jens Nilsson, Sebastian Riedel, and Deniz Yuret. 2007. The CoNLL 2007 shared task on dependency parsing. In *Proceedings of the CoNLL Shared Task Session of EMNLP-CoNLL 2007*, pages 915–932, Prague, Czech Republic.

Joakim Nivre, Cristina Bosco, Jinho Choi, Marie-Catherine de Marneffe, Timothy Dozat, Richárd Farkas, Jennifer Foster, Filip Ginter, Yoav Goldberg, Jan Hajič, Jenna Kanerva, Veronika Laippala, Alessandro Lenci, Teresa Lynn, Christopher Manning, Ryan McDonald, Anna Missilä, Simonetta Montemagni, Slav Petrov, Sampo Pyysalo, Natalia Silveira, Maria Simi, Aaron Smith, Reut Tsarfaty, Veronika Vincze, and Daniel Zeman. 2015. Universal dependencies 1.0.

Levent Özgür and Tunga Güngör. 2010. Text classification with the support of pruned dependency patterns. *Pattern Recognition Letter*, 31:1598–1607.

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2012. A universal part-of-speech tagset. In *Proceedings of the Eight International Conference on Language Resources and Evaluation (LREC'12)*, Istanbul, Turkey.

Anders Søgaard. 2011. Data point selection for cross-language adaptation of dependency parsers. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies: Short Papers - Volume 2*, HLT '11, pages 682–686.

Oscar Täckström, Ryan McDonald, and Jakob Uszkoreit. 2012. Cross-lingual word clusters for direct transfer of linguistic structure. In *Proceedings of the 2012 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, NAACL HLT '12, pages 477–487.

Oscar Täckström, Ryan McDonald, and Joakim Nivre. 2013. Target language adaptation of discriminative transfer parsers. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1061–1071, Atlanta, Georgia.

Joseph Turian, Lev-Arie Ratinov, and Yoshua Bengio. 2010. Word representations: A simple and general method for semi-supervised learning. In *Proceedings of the 48th Annual Meeting of the Association for Computational Linguistics*, pages 384–394, Uppsala, Sweden.

Min Xiao and Yuhong Guo, 2014. *Proceedings of the Eighteenth Conference on Computational Natural Language Learning*, chapter Distributed Word Representation Learning for Cross-Lingual Dependency Parsing, pages 119–129.

Peng Xu, Jaeho Kang, Michael Ringgaard, and Franz Och. 2009. Using a dependency parser to improve smt for subject-object-verb languages. In *Proceedings of Human Language Technologies: The 2009 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 245–253, Boulder, Colorado.

Daniel Zeman, Univerzita Karlova, and Philip Resnik. 2008. Cross-language parser adaptation between related languages. In *In IJCNLP-08 Workshop on NLP for Less Privileged Languages*, pages 35–42.

Will Y. Zou, Richard Socher, Daniel Cer, and Christopher D. Manning. 2013. Bilingual word embeddings for phrase-based machine translation. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1393–1398, Seattle, Washington, USA.