Unsupervised Part-of-Speech Tagging with Bilingual Graph-Based Projections

Dipanjan Das*
Carnegie Mellon University
Pittsburgh, PA 15213, USA
dipanjan@cs.cmu.edu

Slav Petrov
Google Research
New York, NY 10011, USA
slav@google.com

Abstract

We describe a novel approach for inducing unsupervised part-of-speech taggers for languages that have no labeled training data, but have translated text in a resource-rich language. Our method does not assume any knowledge about the target language (in particular no tagging dictionary is assumed), making it applicable to a wide array of resource-poor languages. We use graph-based label propagation for cross-lingual knowledge transfer and use the projected labels as features in an unsupervised model (Berg-Kirkpatrick et al., 2010). Across eight European languages, our approach results in an average absolute improvement of 10.4% over a state-of-the-art baseline, and 16.7% over vanilla hidden Markov models induced with the Expectation Maximization algorithm.

1 Introduction

Supervised learning approaches have advanced the state-of-the-art on a variety of tasks in natural language processing, resulting in highly accurate systems. Supervised part-of-speech (POS) taggers, for example, approach the level of inter-annotator agreement (Shen et al., 2007, 97.3% accuracy for English). However, supervised methods rely on labeled training data, which is time-consuming and expensive to generate. Unsupervised learning approaches appear to be a natural solution to this problem, as they require only unannotated text for training models. Unfortunately, the best completely unsupervised English POS tagger (that does not make use of a tagging dictionary) reaches only 76.1% accuracy (Christodoulopoulos et al., 2010), making its practical usability questionable at best.

To bridge this gap, we consider a practically motivated scenario, in which we want to leverage existing resources from a resource-rich language (like English) when building tools for resource-poor foreign languages.[1] We assume that absolutely no labeled training data is available for the foreign language of interest, but that we have access to parallel data with a resource-rich language. This scenario is applicable to a large set of languages and has been considered by a number of authors in the past (Alshawi et al., 2000; Xi and Hwa, 2005; Ganchev et al., 2009). Naseem et al. (2009) and Snyder et al. (2009) study related but different multilingual grammar and tagger induction tasks, where it is assumed that no labeled data at all is available.

Our work is closest to that of Yarowsky and Ngai (2001), but differs in two important ways. First, we use a novel graph-based framework for projecting syntactic information across language boundaries. To this end, we construct a bilingual graph over word types to establish a connection between the two languages (§3), and then use graph label propagation to project syntactic information from English to the foreign language (§4). Second, we treat the projected labels as features in an unsuper-

*This research was carried out during an internship at Google Research.

[1] For simplicity of exposition we refer to the resource-poor language as the "foreign language." Similarly, we use English as the resource-rich language, but any other language with labeled resources could be used instead.

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 600–609,
Portland, Oregon, June 19-24, 2011.

vised model (§5), rather than using them directly for supervised training. To make the projection practical, we rely on the twelve *universal part-of-speech tags* of Petrov et al. (2011). Syntactic universals are a well studied concept in linguistics (Carnie, 2002; Newmeyer, 2005), and were recently used in similar form by Naseem et al. (2010) for multilingual grammar induction. Because there might be some controversy about the exact definitions of such universals, this set of coarse-grained POS categories is defined operationally, by collapsing language (or treebank) specific distinctions to a set of categories that exists across all languages. These universal POS categories not only facilitate the transfer of POS information from one language to another, but also relieve us from using controversial evaluation metrics,[2] by establishing a direct correspondence between the induced hidden states in the foreign language and the observed English labels.

We evaluate our approach on eight European languages (§6), and show that both our contributions provide consistent and statistically significant improvements. Our final average POS tagging accuracy of 83.4% compares very favorably to the average accuracy of Berg-Kirkpatrick et al.'s monolingual unsupervised state-of-the-art model (73.0%), and considerably bridges the gap to fully supervised POS tagging performance (96.6%).

2 Approach Overview

The focus of this work is on building POS taggers for foreign languages, assuming that we have an English POS tagger and some parallel text between the two languages. Central to our approach (see Algorithm 1) is a bilingual similarity graph built from a sentence-aligned parallel corpus. As discussed in more detail in §3, we use two types of vertices in our graph: on the foreign language side vertices correspond to trigram types, while the vertices on the English side are individual word types. Graph construction does not require any labeled data, but makes use of two similarity functions. The edge weights between the foreign language trigrams are computed using a co-occurence based similarity function, designed to indicate how syntactically similar the middle words of the connected trigrams are (§3.2). To establish a soft correspondence between the two languages, we use a second similarity function, which leverages standard unsupervised word alignment statistics (§3.3).[3]

Algorithm 1 Bilingual POS Induction

Require: Parallel English and foreign language data $\mathcal{D}^e$ and $\mathcal{D}^f$, unlabeled foreign training data Γ^f; English tagger.

Ensure: Θ^f, a set of parameters learned using a constrained unsupervised model (§5).

1: $\mathcal{D}^{e \leftrightarrow f} \leftarrow \textsf{word-align-bitext}(\mathcal{D}^e, \mathcal{D}^f)$
2: $\widehat{\mathcal{D}^e} \leftarrow \textsf{pos-tag-supervised}(\mathcal{D}^e)$
3: $\mathcal{A} \leftarrow \textsf{extract-alignments}(\mathcal{D}^{e \leftrightarrow f}, \widehat{\mathcal{D}^e})$
4: $\mathsf{G} \leftarrow \textsf{construct-graph}(\Gamma^f, \mathcal{D}^f, \mathcal{A})$
5: $\tilde{\mathsf{G}} \leftarrow \textsf{graph-propagate}(\mathsf{G})$
6: $\Delta \leftarrow \textsf{extract-word-constraints}(\tilde{\mathsf{G}})$
7: $\Theta^f \leftarrow \textsf{pos-induce-constrained}(\Gamma^f, \Delta)$
8: Return Θ^f

Since we have no labeled foreign data, our goal is to project syntactic information from the English side to the foreign side. To initialize the graph we tag the English side of the parallel text using a supervised model. By aggregating the POS labels of the English tokens to types, we can generate label distributions for the English vertices. Label propagation can then be used to transfer the labels to the *peripheral* foreign vertices (i.e. the ones adjacent to the English vertices) first, and then among all of the foreign vertices (§4). The POS distributions over the foreign trigram types are used as features to learn a better unsupervised POS tagger (§5). The following three sections elaborate these different stages is more detail.

3 Graph Construction

In graph-based learning approaches one constructs a graph whose vertices are labeled and unlabeled examples, and whose weighted edges encode the degree to which the examples they link have the same label (Zhu et al., 2003). Graph construction for structured prediction problems such as POS tagging is non-trivial: on the one hand, using individual words as the vertices throws away the context

[2] See Christodoulopoulos et al. (2010) for a discussion of metrics for evaluating unsupervised POS induction systems.

[3] The word alignment methods do not use POS information.

necessary for disambiguation; on the other hand, it is unclear how to define (sequence) similarity if the vertices correspond to entire sentences. Altun et al. (2005) proposed a technique that uses graph based similarity between labeled and unlabeled parts of structured data in a discriminative framework for semi-supervised learning. More recently, Subramanya et al. (2010) defined a graph over the cliques in an underlying structured prediction model. They considered a semi-supervised POS tagging scenario and showed that one can use a graph over trigram types, and edge weights based on distributional similarity, to improve a supervised conditional random field tagger.

3.1 Graph Vertices

We extend Subramanya et al.'s intuitions to our bilingual setup. Because the information flow in our graph is asymmetric (from English to the foreign language), we use different types of vertices for each language. The foreign language vertices (denoted by V_f) correspond to foreign trigram types, exactly as in Subramanya et al. (2010). On the English side, however, the vertices (denoted by V_e) correspond to word types. Because all English vertices are going to be labeled, we do not need to disambiguate them by embedding them in trigrams. Furthermore, we do not connect the English vertices to each other, but only to foreign language vertices.[4]

The graph vertices are extracted from the different sides of a parallel corpus ($\mathcal{D}^e$, $\mathcal{D}^f$) and an additional unlabeled monolingual foreign corpus Γ^f, which will be used later for training. We use two different similarity functions to define the edge weights among the foreign vertices and between vertices from different languages.

3.2 Monolingual Similarity Function

Our monolingual similarity function (for connecting pairs of foreign trigram types) is the same as the one used by Subramanya et al. (2010). We briefly review it here for completeness. We define a symmetric similarity function $\mathrm{K}(u_i, u_j)$ over two foreign language vertices $u_i, u_j \in V_f$ based on the co-occurrence statistics of the nine feature concepts given in Table 1. Each feature concept is akin to a random variable and its occurrence in the text corresponds to a particular instantiation of that random variable. For each trigram type $x_2\ x_3\ x_4$ in a sequence $x_1\ x_2\ x_3\ x_4\ x_5$, we count how many times that trigram type co-occurs with the different instantiations of each concept, and compute the point-wise mutual information (PMI) between the two.[5] The similarity between two trigram types is given by summing over the PMI values over feature instantiations that they have in common. This is similar to stacking the different feature instantiations into long (sparse) vectors and computing the cosine similarity between them. Finally, note that while most feature concepts are lexicalized, others, such as the suffix concept, are not.

Description	Feature
Trigram + Context	$x_1\ x_2\ x_3\ x_4\ x_5$
Trigram	$x_2\ x_3\ x_4$
Left Context	$x_1\ x_2$
Right Context	$x_4\ x_5$
Center Word	x_3
Trigram − Center Word	$x_2\ x_4$
Left Word + Right Context	$x_2\ x_4\ x_5$
Left Context + Right Word	$x_1\ x_2\ x_4$
Suffix	HasSuffix(x_3)

Table 1: Various features used for computing edge weights between foreign trigram types.

Given this similarity function, we define a nearest neighbor graph, where the edge weight for the n most similar vertices is set to the value of the similarity function and to 0 for all other vertices. We use $\mathcal{N}(u)$ to denote the neighborhood of vertex u, and fixed $n = 5$ in our experiments.

3.3 Bilingual Similarity Function

To define a similarity function between the English and the foreign vertices, we rely on high-confidence word alignments. Since our graph is built from a parallel corpus, we can use standard word alignment techniques to align the English sentences $\mathcal{D}^e$

[4] This is because we are primarily interested in learning foreign language taggers, rather than improving supervised English taggers. Note, however, that it would be possible to use our graph-based framework also for completely unsupervised POS induction in both languages, similar to Snyder et al. (2009).

[5] Note that many combinations are impossible giving a PMI value of 0; e.g., when the trigram type and the feature instantiation don't have words in common.

and their foreign language translations $\mathcal{D}^f$.[6] Label propagation in the graph will provide coverage and high recall, and we therefore extract only intersected high-confidence (> 0.9) alignments $\mathcal{D}^{e \leftrightarrow f}$.

Based on these high-confidence alignments we can extract tuples of the form $[u \leftrightarrow v]$, where u is a foreign trigram type, whose middle word aligns to an English word type v. Our bilingual similarity function then sets the edge weights in proportion to these tuple counts.

3.4 Graph Initialization

So far the graph has been completely unlabeled. To initialize the graph for label propagation we use a supervised English tagger to label the English side of the bitext.[7] We then simply count the individual labels of the English tokens and normalize the counts to produce tag distributions over English word types. These tag distributions are used to initialize the label distributions over the English vertices in the graph. Note that since all English vertices were extracted from the parallel text, we will have an initial label distribution for all vertices in V_e.

3.5 Graph Example

A very small excerpt from an Italian-English graph is shown in Figure 1. As one can see, only the trigrams [suo **incarceramento** ,], [suo **iter** ,] and [suo **carattere** ,] are connected to English words. In this particular case, all English vertices are labeled as nouns by the supervised tagger. In general, the neighborhoods can be more diverse and we allow a soft label distribution over the vertices. It is worth noting that the middle words of the Italian trigrams are nouns too, which exhibits the fact that the similarity metric connects types having the same syntactic category. In the label propagation stage, we propagate the automatic English tags to the aligned Italian trigram types, followed by further propagation solely among the Italian vertices.

[6] We ran six iterations of IBM Model 1 (Brown et al., 1993), followed by six iterations of the HMM model (Vogel et al., 1996) in both directions.

[7] We used a tagger based on a trigram Markov model (Brants, 2000) trained on the Wall Street Journal portion of the Penn Treebank (Marcus et al., 1993), for its fast speed and reasonable accuracy (96.7% on sections 22-24 of the treebank, but presumably much lower on the (out-of-domain) parallel corpus).

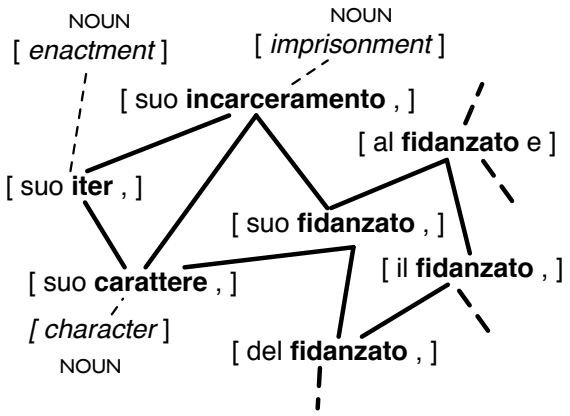

Figure 1: An excerpt from the graph for Italian. Three of the Italian vertices are connected to an automatically labeled English vertex. Label propagation is used to propagate these tags inwards and results in tag distributions for the middle word of each Italian trigram.

4 POS Projection

Given the bilingual graph described in the previous section, we can use label propagation to project the English POS labels to the foreign language. We use label propagation in two stages to generate soft labels on all the vertices in the graph. In the first stage, we run a single step of label propagation, which transfers the label distributions from the English vertices to the connected foreign language vertices (say, V_f^l) at the periphery of the graph. Note that because we extracted only high-confidence alignments, many foreign vertices will not be connected to any English vertices. This stage of label propagation results in a tag distribution r_i over labels y, which encodes the proportion of times the middle word of $u_i \in V_f$ aligns to English words v_y tagged with label y:

$$\mathrm{r}_i(y) = \frac{\sum_{v_y} \#[u_i \leftrightarrow v_y]}{\sum_{y'} \sum_{v_{y'}} \#[u_i \leftrightarrow v_{y'}]} \quad (1)$$

The second stage consists of running traditional label propagation to propagate labels from these peripheral vertices V_f^l to all foreign language vertices

in the graph, optimizing the following objective:

$$\begin{aligned}\mathcal{C}(\mathrm{q}) &= \sum_{u_i \in V_f \setminus V_f^l, u_j \in \mathcal{N}(u_i)} w_{ij} \|\mathrm{q}_i - \mathrm{q}_j\|^2 \\ &+ \nu \sum_{u_i \in V_f \setminus V_f^l} \|\mathrm{q}_i - U\|^2 \\ \text{s.t.} \quad & \sum_y \mathrm{q}_i(y) = 1 \; \forall u_i \\ & \mathrm{q}_i(y) \geq 0 \; \forall u_i, y \\ & \mathrm{q}_i = \mathrm{r}_i \; \forall u_i \in V_f^l \end{aligned} \tag{2}$$

where the q_i ($i = 1, \ldots, |V_f|$) are the label distributions over the foreign language vertices and μ and ν are hyperparameters that we discuss in §6.4. We use a squared loss to penalize neighboring vertices that have different label distributions: $\|\mathrm{q}_i - \mathrm{q}_j\|^2 = \sum_y (\mathrm{q}_i(y) - \mathrm{q}_j(y))^2$, and additionally regularize the label distributions towards the uniform distribution U over all possible labels $\mathcal{Y}$. It can be shown that this objective is convex in q.

The first term in the objective function is the graph smoothness regularizer which encourages the distributions of similar vertices (large w_{ij}) to be similar. The second term is a regularizer and encourages all type marginals to be uniform to the extent that is allowed by the first two terms (cf. maximum entropy principle). If an unlabeled vertex does not have a path to any labeled vertex, this term ensures that the converged marginal for this vertex will be uniform over all tags, allowing the middle word of such an unlabeled vertex to take on any of the possible tags.

While it is possible to derive a closed form solution for this convex objective function, it would require the inversion of a matrix of order $|V_f|$. Instead, we resort to an iterative update based method. We formulate the update as follows:

$$\mathrm{q}_i^{(m)}(y) = \begin{cases} \mathrm{r}_i(y) & \text{if } u_i \in V_f^l \\ \dfrac{\gamma_i(y)}{\kappa_i} & \text{otherwise} \end{cases} \tag{3}$$

where $\forall u_i \in V_f \setminus V_f^l$, $\gamma_i(y)$ and κ_i are defined as:

$$\gamma_i(y) = \sum_{u_j \in \mathcal{N}(u_i)} w_{ij} \mathrm{q}_j^{(m-1)}(y) + \nu \, U(y) \tag{4}$$

$$\kappa_i = \nu + \sum_{u_j \in \mathcal{N}(u_i)} w_{ij} \tag{5}$$

We ran this procedure for 10 iterations.

5 POS Induction

After running label propagation (LP), we compute tag probabilities for foreign word types x by marginalizing the POS tag distributions of foreign trigrams $u_i = x_- \; x \; x_+$ over the left and right context words:

$$p(y|x) = \frac{\sum_{x_-, x_+} \mathrm{q}_i(y)}{\sum_{x_-, x_+, y'} \mathrm{q}_i(y')} \tag{6}$$

We then extract a set of possible tags $\mathtt{t}_x(y)$ by eliminating labels whose probability is below a threshold value τ:

$$\mathtt{t}_x(y) = \begin{cases} 1 & \text{if } p(y|x) \geq \tau \\ 0 & \text{otherwise} \end{cases} \tag{7}$$

We describe how we choose τ in §6.4. This vector $\mathtt{t}_x$ is constructed for every word in the foreign vocabulary and will be used to provide features for the unsupervised foreign language POS tagger.

We develop our POS induction model based on the feature-based HMM of Berg-Kirkpatrick et al. (2010). For a sentence $\mathbf{x}$ and a state sequence $\mathbf{z}$, a first order Markov model defines a distribution:

$$\begin{aligned} P_\Theta(\mathbf{X} = \mathbf{x}, \mathbf{Z} = \mathbf{z}) &= P_\Theta(Z_1 = z_1) \cdot \\ \prod_{i=1}^{|\mathbf{x}|} & \underbrace{P_\Theta(Z_{i+1} = z_{i+1} \mid Z_i = z_i)}_{\text{transition}} \cdot \\ & \underbrace{P_\Theta(X_i = x_i \mid Z_i = z_i)}_{\text{emission}} \end{aligned} \tag{8}$$

In a traditional Markov model, the emission distribution $P_\Theta(X_i = x_i \mid Z_i = z_i)$ is a set of multinomials. The feature-based model replaces the emission distribution with a log-linear model, such that:

$$P_\Theta(X = x \mid Z = z) = \frac{\exp \Theta^\top \mathbf{f}(x, z)}{\sum_{x' \in \text{Val}(X)} \exp \Theta^\top \mathbf{f}(x', z)} \tag{9}$$

where $\text{Val}(X)$ corresponds to the entire vocabulary. This locally normalized log-linear model can look at various aspects of the observation x, incorporating overlapping features of the observation. In our experiments, we used the same set of features as Berg-Kirkpatrick et al. (2010): an indicator feature based

on the word identity x, features checking whether x contains digits or hyphens, whether the first letter of x is upper case, and suffix features up to length 3. All features were conjoined with the state z.

We trained this model by optimizing the following objective function:

$$\mathcal{L}(\Theta) = \sum_{i=1}^{N} \log \sum_{\mathbf{z}} P_\Theta(\mathbf{X} = \mathbf{x}^{(i)}, \mathbf{Z} = \mathbf{z}^{(i)}) - C\|\Theta\|_2^2 \quad (10)$$

Note that this involves marginalizing out all possible state configurations $\mathbf{z}$ for a sentence $\mathbf{x}$, resulting in a non-convex objective. To optimize this function, we used L-BFGS, a quasi-Newton method (Liu and Nocedal, 1989). For English POS tagging, Berg-Kirkpatrick et al. (2010) found that this direct gradient method performed better (>7% absolute accuracy) than using a feature-enhanced modification of the Expectation-Maximization (EM) algorithm (Dempster et al., 1977).[8] Moreover, this route of optimization outperformed a vanilla HMM trained with EM by 12%.

We adopted this state-of-the-art model because it makes it easy to experiment with various ways of incorporating our novel *constraint* feature into the log-linear emission model. This feature $\mathsf{f_t}$ incorporates information from the smoothed graph and prunes hidden states that are inconsistent with the thresholded vector t_x. The function $\lambda : \mathsf{F} \rightarrow \mathsf{C}$ maps from the language specific fine-grained tagset F to the coarser universal tagset C and is described in detail in §6.2:

$$\mathsf{f_t}(x, z) = \log(\mathsf{t}_x(y)), \text{ if } \lambda(z) = y \quad (11)$$

Note that when $\mathsf{t}_x(y) = 1$ the feature value is 0 and has no effect on the model, while its value is $-\infty$ when $\mathsf{t}_x(y) = 0$ and constrains the HMM's state space. This formulation of the constraint feature is equivalent to the use of a tagging dictionary extracted from the graph using a threshold τ on the posterior distribution of tags for a given word type (Eq. 7). It would have therefore also been possible to use the integer programming (IP) based approach of Ravi and Knight (2009) instead of the feature-HMM for POS induction on the foreign side. However, we do not explore this possibility in the current work.

[8] See §3.1 of Berg-Kirkpatrick et al. (2010) for more details about their modification of EM, and how gradients are computed for L-BFGS.

6 Experiments and Results

Before presenting our results, we describe the datasets that we used, as well as two baselines.

6.1 Datasets

We utilized two kinds of datasets in our experiments: (i) monolingual treebanks[9] and (ii) large amounts of parallel text with English on one side. The availability of these resources guided our selection of foreign languages. For monolingual treebank data we relied on the CoNLL-X and CoNLL-2007 shared tasks on dependency parsing (Buchholz and Marsi, 2006; Nivre et al., 2007). The parallel data came from the Europarl corpus (Koehn, 2005) and the ODS United Nations dataset (UN, 2006). Taking the intersection of languages in these resources, and selecting languages with large amounts of parallel data, yields the following set of eight Indo-European languages: Danish, Dutch, German, Greek, Italian, Portuguese, Spanish and Swedish.

Of course, we are primarily interested in applying our techniques to languages for which no labeled resources are available. However, we needed to restrict ourselves to these languages in order to be able to evaluate the performance of our approach. We paid particular attention to minimize the number of free parameters, and used the same hyperparameters for all language pairs, rather than attempting language-specific tuning. We hope that this will allow practitioners to apply our approach directly to languages for which no resources are available.

6.2 Part-of-Speech Tagset and HMM States

We use the universal POS tagset of Petrov et al. (2011) in our experiments.[10] This set C consists of the following 12 coarse-grained tags: NOUN (nouns), VERB (verbs), ADJ (adjectives), ADV (adverbs), PRON (pronouns), DET (determiners), ADP (prepositions or postpositions), NUM (numerals), CONJ (conjunctions), PRT (particles), PUNC

[9] We extracted only the words and their POS tags from the treebanks.

[10] Available at http://code.google.com/p/universal-pos-tags/.

(punctuation marks) and X (a catch-all for other categories such as abbreviations or foreign words). While there might be some controversy about the exact definition of such a tagset, these 12 categories cover the most frequent part-of-speech and exist in one form or another in all of the languages that we studied.

For each language under consideration, Petrov et al. (2011) provide a mapping λ from the fine-grained language specific POS tags in the foreign treebank to the universal POS tags. The supervised POS tagging accuracies (on this tagset) are shown in the last row of Table 2. The taggers were trained on datasets labeled with the universal tags.

The number of latent HMM states for each language in our experiments was set to the number of fine tags in the language's treebank. In other words, the set of hidden states F was chosen to be the fine set of treebank tags. Therefore, the number of fine tags varied across languages for our experiments; however, one could as well have fixed the set of HMM states to be a constant across languages, and created one mapping to the universal POS tagset.

6.3 Various Models

To provide a thorough analysis, we evaluated three baselines and two oracles in addition to two variants of our graph-based approach. We were intentionally lenient with our baselines:

- **EM-HMM:** A traditional HMM baseline, with multinomial emission and transition distributions estimated by the Expectation Maximization algorithm. We evaluated POS tagging accuracy using the lenient many-to-1 evaluation approach (Johnson, 2007).
- **Feature-HMM:** The vanilla feature-HMM of Berg-Kirkpatrick et al. (2010) (i.e. no additional constraint feature) served as a second baseline. Model parameters were estimated with L-BFGS and evaluation again used a greedy many-to-1 mapping.
- **Projection:** Our third baseline incorporates bilingual information by projecting POS tags directly across alignments in the parallel data. For unaligned words, we set the tag to the most frequent tag in the corresponding treebank. For each language, we took the same number of sentences from the bitext as there are in its treebank, and trained a *supervised* feature-HMM. This can be seen as a rough approximation of Yarowsky and Ngai (2001).

We tried two versions of our graph-based approach:

- **No LP:** Our first version takes advantage of our bilingual graph, but extracts the constraint feature after the first stage of label propagation (Eq. 1). Because many foreign word types are not aligned to an English word (see Table 3), and we do not run label propagation on the foreign side, we expect the projected information to have less coverage. Furthermore we expect the label distributions on the foreign to be fairly noisy, because the graph constraints have not been taken into account yet.
- **With LP:** Our full model uses both stages of label propagation (Eq. 2) before extracting the constraint features. As a result, we are able to extract the constraint feature for all foreign word types and furthermore expect the projected tag distributions to be smoother and more stable.

Our oracles took advantage of the labeled treebanks:

- **TB Dictionary:** We extracted tagging dictionaries from the treebanks and and used them as constraint features in the feature-based HMM. Evaluation was done using the prespecified mappings.
- **Supervised:** We trained the supervised model of Brants (2000) on the original treebanks and mapped the language-specific tags to the universal tags for evaluation.

6.4 Experimental Setup

While we tried to minimize the number of free parameters in our model, there are a few hyperparameters that need to be set. Fortunately, performance was stable across various values, and we were able to use the same hyperparameters for all languages.

We used $C = 1.0$ as the L_2 regularization constant in (Eq. 10) and trained both EM and L-BFGS for 1000 iterations. When extracting the vector

	Model	Danish	Dutch	German	Greek	Italian	Portuguese	Spanish	Swedish	**Avg**
baselines	EM-HMM	68.7	57.0	75.9	65.8	63.7	62.9	71.5	68.4	66.7
	Feature-HMM	69.1	65.1	81.3	71.8	68.1	78.4	80.2	70.1	73.0
	Projection	73.6	77.0	83.2	79.3	79.7	82.6	80.1	74.7	78.8
our approach	No LP	79.0	78.8	82.4	76.3	84.8	87.0	82.8	79.4	81.3
	With LP	**83.2**	**79.5**	82.8	**82.5**	**86.8**	**87.9**	**84.2**	**80.5**	83.4
oracles	TB Dictionary	*93.1*	*94.7*	*93.5*	*96.6*	*96.4*	*94.0*	*95.8*	*85.5*	*93.7*
	Supervised	*96.9*	*94.9*	*98.2*	*97.8*	*95.8*	*97.2*	*96.8*	*94.8*	*96.6*

Table 2: Part-of-speech tagging accuracies for various baselines and oracles, as well as our approach. "**Avg**" denotes macro-average across the eight languages.

t_x used to compute the constraint feature from the graph, we tried three threshold values for τ (see Eq. 7). Because we don't have a separate development set, we used the training set to select among them and found 0.2 to work slightly better than 0.1 and 0.3. For seven out of eight languages a threshold of 0.2 gave the best results for our final model, which indicates that for languages without any validation set, $\tau = 0.2$ can be used. For graph propagation, the hyperparameter ν was set to 2×10^{-6} and was not tuned. The graph was constructed using 2 million trigrams; we chose these by truncating the parallel datasets up to the number of sentence pairs that contained 2 million trigrams.

6.5 Results

Table 2 shows our complete set of results. As expected, the vanilla HMM trained with EM performs the worst. The feature-HMM model works better for all languages, generalizing the results achieved for English by Berg-Kirkpatrick et al. (2010). Our "Projection" baseline is able to benefit from the bilingual information and greatly improves upon the monolingual baselines, but falls short of the "No LP" model by 2.5% on an average. The "No LP" model does not outperform direct projection for German and Greek, but performs better for six out of eight languages. Overall, it gives improvements ranging from 1.1% for German to 14.7% for Italian, for an average improvement of 8.3% over the unsupervised feature-HMM model. For comparison, the completely unsupervised feature-HMM baseline accuracy on the universal POS tags for English is 79.4%, and goes up to 88.7% with a treebank dictionary.

Our full model ("With LP") outperforms the unsupervised baselines and the "No LP" setting for all languages. It falls short of the "Projection" baseline for German, but is statistically indistinguishable in terms of accuracy. As indicated by bolding, for seven out of eight languages the improvements of the "With LP" setting are statistically significant with respect to the other models, including the "No LP" setting.[11] Overall, it performs 10.4% better than the hitherto state-of-the-art feature-HMM baseline, and 4.6% better than direct projection, when we macro-average the accuracy over all languages.

6.6 Discussion

Our full model outperforms the "No LP" setting because it has better vocabulary coverage and allows the extraction of a larger set of constraint features. We tabulate this increase in Table 3. For all languages, the vocabulary sizes increase by several thousand words. Although the tag distributions of the foreign words (Eq. 6) are noisy, the results confirm that label propagation within the foreign language part of the graph adds significant quality for every language.

Figure 2 shows an excerpt of a sentence from the Italian test set and the tags assigned by four different models, as well as the gold tags. While the first three models get three to four tags wrong, our best model gets only one word wrong and is the most accurate among the four models for this example. Examining the word *fidanzato* for the "No LP" and "With LP" models is particularly instructive. As Figure 1 shows, this word has no high-confidence alignment in the Italian-English bitext. As a result, its POS tag needs to be induced in the "No LP" case, while the

[11] A word level paired-t-test is significant at $p < 0.01$ for Danish, Greek, Italian, Portuguese, Spanish and Swedish, and $p < 0.05$ for Dutch.

	si	trovava	in	un	parco	con	il	fidanzato	Paolo	F.	,	27	anni	,	rappresentante
EM-HMM:	*CONJ*	*NOUN*	*DET*	DET	NOUN	ADP	DET	NOUN	.	NOUN	.	NUM	NOUN	.	NOUN
Feature-HMM:	PRON	VERB	ADP	DET	NOUN	*CONJ*	DET	NOUN	NOUN	NOUN	.	*ADP*	NOUN	.	*VERB*
No LP:	*VERB*	VERB	ADP	DET	NOUN	ADP	DET	*ADJ*	NOUN	*ADJ*	.	NUM	NOUN	.	NOUN
With LP:	*VERB*	VERB	ADP	DET	NOUN	ADP	DET	NOUN	NOUN	NOUN	.	NUM	NOUN	.	NOUN
Gold:	PRON	VERB	ADP	DET	NOUN	ADP	DET	NOUN	NOUN	NOUN	.	NUM	NOUN	.	NOUN

Figure 2: Tags produced by the different models along with the reference set of tags for a part of a sentence from the Italian test set. Italicized tags denote incorrect labels.

Language	**# words with constraints**	
	"No LP"	"With LP"
Danish	88,240	128, 391
Dutch	51,169	74,892
German	59,534	107,249
Greek	90,231	114,002
Italian	48,904	62,461
Portuguese	46,787	65,737
Spanish	72,215	82,459
Swedish	70,181	88,454

Table 3: Size of the vocabularies for the "No LP" and "With LP" models for which we can impose constraints.

correct tag is available as a constraint feature in the "With LP" case.

7 Conclusion

We have shown the efficacy of graph-based label propagation for projecting part-of-speech information across languages. Because we are interested in applying our techniques to languages for which no labeled resources are available, we paid particular attention to minimize the number of free parameters and used the same hyperparameters for all language pairs. Our results suggest that it is possible to learn accurate POS taggers for languages which do not have any annotated data, but have translations into a resource-rich language. Our results outperform strong unsupervised baselines as well as approaches that rely on direct projections, and bridge the gap between purely supervised and unsupervised POS tagging models.

Acknowledgements

We would like to thank Ryan McDonald for numerous discussions on this topic. We would also like to thank Amarnag Subramanya for helping us with the implementation of label propagation and Shankar Kumar for access to the parallel data. Finally, we thank Kuzman Ganchev and the three anonymous reviewers for helpful suggestions and comments on earlier drafts of this paper.

References

Hiyan Alshawi, Srinivas Bangalore, and Shona Douglas. 2000. Head-transducer models for speech translation and their automatic acquisition from bilingual data. *Machine Translation*, 15.

Yasemin Altun, David McAllester, and Mikhail Belkin. 2005. Maximum margin semi-supervised learning for structured variables. In *Proc. of NIPS*.

Taylor Berg-Kirkpatrick, Alexandre B. Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Proc. of NAACL-HLT*.

Thorsten Brants. 2000. TnT - a statistical part-of-speech tagger. In *Proc. of ANLP*.

Peter F. Brown, Vincent J. Della Pietra, Stephen A. Della Pietra, and Robert L. Mercer. 1993. The mathematics of statistical machine translation: parameter estimation. *Computational Linguistics*, 19.

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

Andrew Carnie. 2002. *Syntax: A Generative Introduction (Introducing Linguistics)*. Blackwell Publishing.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2010. Two decades of unsupervised POS induction: How far have we come? In *Proc. of EMNLP*.

Arthur P. Dempster, Nan M. Laird, and Donald B. Rubin. 1977. Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, 39.

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

Mark Johnson. 2007. Why doesn't EM find good HMM POS-taggers? In *Proc. of EMNLP-CoNLL*.

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

Dong C. Liu and Jorge Nocedal. 1989. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45.

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

Tahira Naseem, Benjamin Snyder, Jacob Eisenstein, and Regina Barzilay. 2009. Multilingual part-of-speech tagging: Two unsupervised approaches. *JAIR*, 36.

Tahira Naseem, Harr Chen, Regina Barzilay, and Mark Johnson. 2010. Using universal linguistic knowledge to guide grammar induction. In *Proc. of EMNLP*.

Frederick J. Newmeyer. 2005. *Possible and Probable Languages: A Generative Perspective on Linguistic Typology.* Oxford University Press.

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 CoNLL.*

Slav Petrov, Dipanjan Das, and Ryan McDonald. 2011. A universal part-of-speech tagset. *ArXiv:1104.2086*.

Sujith Ravi and Kevin Knight. 2009. Minimized models for unsupervised part-of-speech tagging. In *Proc. of ACL-IJCNLP*.

Libin Shen, Giorgio Satta, and Aravind Joshi. 2007. Guided learning for bidirectional sequence classification. In *Proc. of ACL.*

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

Amar Subramanya, Slav Petrov, and Fernando Pereira. 2010. Efficient graph-based semi-supervised learning of structured tagging models. In *Proc. of EMNLP*.

UN. 2006. ODS UN parallel corpus.

Stephan Vogel, Hermann Ney, and Christoph Tillmann. 1996. HMM-based word alignment in statistical translation. In *Proc. of COLING*.

Chenhai Xi and Rebecca Hwa. 2005. A backoff model for bootstrapping resources for non-English languages. In *Proc. of HLT-EMNLP*.

David Yarowsky and Grace Ngai. 2001. Inducing multilingual POS taggers and NP bracketers via robust projection across aligned corpora. In *Proc. of NAACL.*

Xiaojin Zhu, Zoubin Ghahramani, and John D. Lafferty. 2003. Semi-supervised learning using gaussian fields and harmonic functions. In *Proc. of ICML.*