Learning to Map into a Universal POS Tagset

Yuan Zhang, Roi Reichart, Regina Barzilay
Massachusetts Institute of Technology
{yuanzh, roiri, regina}@csail.mit.edu

Amir Globerson
The Hebrew University
gamir@cs.huji.ac.il

Abstract

We present an automatic method for mapping *language-specific* part-of-speech tags to a set of *universal tags*. This unified representation plays a crucial role in cross-lingual syntactic transfer of multilingual dependency parsers. Until now, however, such conversion schemes have been created manually. Our central hypothesis is that a valid mapping yields POS annotations with coherent linguistic properties which are consistent across source and target languages. We encode this intuition in an objective function that captures a range of distributional and typological characteristics of the derived mapping. Given the exponential size of the mapping space, we propose a novel method for optimizing over soft mappings, and use entropy regularization to drive those towards hard mappings. Our results demonstrate that automatically induced mappings rival the quality of their manually designed counterparts when evaluated in the context of multilingual parsing.[1]

1 Introduction

In this paper, we explore an automatic method for mapping *language-specific* part-of-speech tags to a *universal tagset*. In multilingual parsing, this unified input representation is required for cross-lingual syntactic transfer. Specifically, the universal tagset annotations enable an unlexicalized parser to capitalize on annotations from one language when learning a model for another.

[1]The source code and data for the work presented in this paper is available at http://groups.csail.mit.edu/rbg/code/unitag/emnlp2012

While the notion of a universal POS tagset is widely accepted, in practice it is hardly ever used for annotation of monolingual resources. In fact, available POS annotations are designed to capture language-specific idiosyncrasies and therefore are substantially more detailed than a coarse universal tagset. To reconcile these cross-lingual annotation differences, a number of mapping schemes have been proposed in the parsing community (Zeman and Resnik, 2008; Petrov et al., 2011; Naseem et al., 2010). In all of these cases, the conversion is performed manually and has to be repeated for each language and annotation scheme anew.

Despite the apparent simplicity, deriving a mapping is by no means easy, even for humans. In fact, the universal tagsets manually induced by Petrov et al. (2011) and by Naseem et al. (2010) disagree on 10% of the tags. An example of such discrepancy is the mapping of the Japanese tag "PVfin" to the universal tag "particle" according to one scheme, and to "verb" according to another. Moreover, the quality of this conversion has a direct implication on the parsing performance. In the Japanese example above, this difference in mapping yields a 6.7% difference in parsing accuracy.

The goal of our work is to induce the mapping for a new language, utilizing existing manually-constructed mappings as training data. The existing mappings developed in the parsing community rely on gold POS tags for the target language. A more realistic scenario is to employ the mapping technique to resource-poor languages where gold POS annotations are lacking. In such cases, a mapping algorithm has to operate over automatically in-

duced clusters on the target language (e.g., using the Brown algorithm) and convert them to universal tags. We are interested in a mapping approach that can effectively handle both gold tags and induced clusters.

Our central hypothesis is that a valid mapping yields POS annotations with coherent linguistic properties which are consistent across languages. Since universal tags play the same linguistic role in source and target languages, we expect similarity in their *global distributional statistics*. Figure 1a shows statistics for two close languages, English and German. We can see that their unigram frequencies on the five most common tags are very close. Other properties concern *POS tag per sentence statistics* – e.g., every sentence has to have at least one verb. Finally, the mappings can be further constrained by *typological properties* of the target language that specify likely tag sequences. This information is readily available even for resource poor language (Haspelmath et al., 2005). For instance, since English and German are prepositional languages, we expect to observe adposition-noun sequences but not the reverse (see Figure 1b for sample sentences). We encode these heterogeneous properties into an objective function that guides the search for the optimal mapping.

Having defined a quality measure for mappings, our goal is to find the optimal mapping. However, such partition optimization problems[2] are NP hard (Garey and Johnson, 1979). A naive approach to the problem is to greedily improve the map, but it turns out that this approach yields poor quality mappings. We therefore develop a method for optimizing over soft mappings, and use entropy regularization to drive those towards hard mappings. We construct the objective in a way that facilitates simple monotonically improving updates corresponding to solving convex optimization problems.

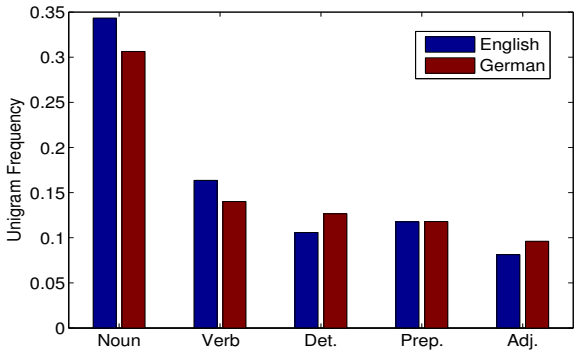
We evaluate our mapping approach on 19 languages that include representatives of Indo-European, Semitic, Basque, Japonic and Turkic families. We measure mapping quality based on the target language parsing accuracy. In addition to considering gold POS tags for the target language, we also evaluate the mapping algorithm on automatically induced POS tags. In all evaluation scenarios, our model consistently rivals the quality of manually induced mappings. We also demonstrate that the proposed inference procedure outperforms greedy methods by a large margin, highlighting the importance of good optimization techniques. We further show that while all characteristics of the mapping contribute to the objective, our largest gain comes from distributional features that capture global statistics. Finally, we establish that the mapping quality has a significant impact on the accuracy of syntactic transfer, which motivates further study of this topic.

[2]Instances of related hard problems are 3-partition and subset-sum.

2 Related Work

Multilingual Parsing Early approaches for multilingual parsing used parallel data to bridge the gap between languages when modeling syntactic transfer. In this setup, finding the mapping between various POS annotation schemes was not essential; instead, the transfer algorithm could induce it directly from the parallel data (Hwa et al., 2005; Xi and Hwa, 2005; Burkett and Klein, 2008). However, more recent transfer approaches relinquish this data requirement, learning to transfer from non-parallel data (Zeman and Resnik, 2008; McDonald et al., 2011; Cohen et al., 2011; Naseem et al., 2010). These approaches assume access to a common input representation in the form of universal tags, which enables the model to connect patterns observed in the source language to their counterparts in the target language.

Despite ongoing efforts to standardize POS tags across languages (e.g., EAGLES initiative (Eynde, 2004)), many corpora are still annotated with language-specific tags. In previous work, their mapping to universal tags was performed manually. Yet, even though some of these mappings have been developed for the same CoNLL dataset (Buchholz and Marsi, 2006; Nivre et al., 2007), they are not identical and yield different parsing performance (Zeman and Resnik, 2008; Petrov et al., 2011; Naseem et al., 2010). The goal of our work is to automate this process and construct mappings that are optimized for performance on downstream tasks (here we focus on parsing). As our results show, we achieve this goal



(a)

-Investors [are appealing] to the Securities and Exchange Commission not to [limit] their access to information [*about* stock purchases] and sales [*by* corporate insiders]

-Einer der sich [*für* den Milliardär] [ausspricht] [ist] Steve Jobs dem Perot [*für* den aufbau] der Computerfirma Next 20 Millionen Dollar [bereitstellte]

(b)

Figure 1: Illustration of similarities in POS tag statistics across languages. (a) The unigram frequency statistics on five tags for two close languages, English and German. (b) Sample sentences in English and German. Verbs are shown in blue, prepositions in red and noun phrases in green. It can be seen that noun phrases follow prepositions.

on a broad range of languages and evaluation scenarios.

Syntactic Category Refinement Our work also relates to work in syntactic category refinement in which POS categories and parse tree non-terminals are refined in order to improve parsing performance (Finkel et al., 2007; Klein and Manning, 2003; Matsuzaki et al., 2005; Petrov et al., 2006; Petrov and Klein, 2007; Liang et al., 2007). Our work differs from these approaches in two ways. First, these methods have been developed in the monolingual setting, while our mapping algorithm is designed for multilingual parsing. Second, these approaches are trained on the syntactic trees of the target language, which enables them to directly link the quality of newly induced categories with the quality of syntactic parsing. In contrast, we are not given trees in the target language. Instead, our model is informed by mappings derived for other languages.

3 Task Formulation

The input to our task consists of a *target* corpus written in a language T, and a set of non-parallel *source* corpora written in languages $\{S_1, \ldots, S_n\}$. In the source corpora, each word is annotated with both a language-specific POS tag and a universal POS tag (Petrov et al., 2011). In the target corpus each word is annotated only with a language-specific POS tag, either gold or automatically induced.

Our goal is to find a map from the set of L_T target language tags to the set of K universal tags. We assume that each language-specific tag is only mapped to one universal tag, which means we never split a language-specific tag and $L_T \geq K$ holds for every language. We represent the map by a matrix A of size $K \times L_T$ where $A(c|f) = 1$ if the target language tag f is mapped to the universal tag c, and $A(c|f) = 0$ otherwise.[3] Note that each column of A should contain a single value of 1. We will later relax the requirement that $A(c|f) \in \{0, 1\}$. A candidate mapping A can be applied to the target language to produce sentences labeled with universal tags.

4 Model

In this section we describe an objective that reflects the quality of an automatic mapping.

Our key insight is that for a good mapping, the statistics over the universal tags should be similar for source and target languages because these tags play the same role cross-linguistically. For example, we should expect the frequency of a particular universal tag to be similar in the source and target languages.

One choice to make when constructing an objective is the source languages to which we want to be similar. It is clear that choosing all languages is not a good idea, since they are not all expected to have distributional properties similar to the target language. There is strong evidence that projecting from single languages can lead to good parsing performance

[3] We use c and f to reflect the fact that universal tags are a coarse version (hence c) of the language specific fine tags (hence f).

(McDonald et al., 2011). Therefore, our strategy is to choose a single source language for comparison. The choice of the source language is based on similarity between typological properties; we describe this in detail in Section 5.

We must also determine which statistical properties we expect to be preserved across languages. Our model utilizes three linguistic phenomena which are consistent across languages: POS tag global distributional statistics, POS tag per sentence statistics, and typology-based ordering statistics. We define each of these below.

4.1 Mapping Characterization

We focus on three categories of mapping properties. For each of the relevant statistics we define a function $F_i(A)$ that has low values if the source and target statistics are similar.

Global distributional statistics: The unigram and bigram statistics of the universal tags are expected to be similar across languages with close typological profiles. We use $p_S(c_1, c_2)$ to denote the bigram distribution over universal tags in the source language, and $p_T(f_1, f_2)$ to denote the bigram distribution over language specific tags in the target language. The bigram distribution over universal tags in the target language depends on A and $p_T(f_1, f_2)$ and is given by:

$$p_T(c_1, c_2; A) = \sum_{f_1, f_2} A(c_1|f_1)A(c_2|f_2)p_T(f_1, f_2) \quad (1)$$

To enforce similarity between source and target distributions, we wish to minimize the KL divergence between the two: [4]

$$F_{bi}(A) = D_{KL}[p_S(c_1, c_2)|p_T(c_1, c_2; A)] \quad (2)$$

We similarly define $F_{uni}(A)$ as the distance between unigram distributions.

Per sentence statistics: Another defining property of POS tags is their average count per sentence. Specifically, we focus on the verb count per sentence, which we expect be similar across languages. To express this constraint, we use $n_v(s, A)$ to denote the number of verbs (i.e., the universal tags corresponding to verbs according to A) in sentence s. This is a linear function of A. We also use $E[n_v(s, A)]$ to denote the average number of verbs per sentence, and $V[n_v(s, A)]$ to denote the variance. We estimate these two statistics from the source language and denote them by E_{Sv}, V_{Sv}. Good mappings are expected to follow these patterns by having a variance upper bounded by V_{Sv} and an average lower bounded by E_{Sv}.[5] This corresponds to minimizing the following objectives:

$$\begin{aligned} F_{Ev}(A) &= \max[0, E_{Sv} - E[n_v(s, A)]] \\ F_{Vv}(A) &= \max[0, V[n_v(s, A)] - V_{Sv}] \end{aligned}$$

Note that the above objectives are convex in A, which will make optimization simpler. We refer to the two terms jointly as $F_{verb}(A)$.

Typology-based ordering statistics: Typological features can be useful for determining the relative order of different tags. If we know that the target language has a particular typological feature, we expect its universal tags to obey the given relative ordering. Specifically, we expect it to agree with ordering statistics for source languages with a similar typology. We consider two such features here. First, in pre-position languages the preposition is followed by the noun phrase. Thus, if T is such a language, we expect the probability of a noun phrase following the adposition to be high, i.e., cross some threshold. Formally, we define C_1 = {noun, adj, num, pron, det} and consider the set of bigram distributions $\mathcal{S}_{\text{pre}}$ that satisfy the following constraint:

$$\sum_{c \in C_1} p_T(\text{adp}, c) \geq a_{\text{pre}} \quad (3)$$

where $a_{\text{pre}} = \sum_{c \in C_1} p_S(\text{adp}, c)$ is calculated from the source language. This constraint set is non-convex in A due to the bilinearity of the bigram term. To simplify optimization[6] we take an

[4] We use the KL divergence because it assigns low weights to infrequent universal tags. Furthermore, this choice results in a simple, EM-like parameter estimation algorithm as discussed in Section 5.

[5] The rationale is that we want to put a lower bound on the number of verbs per sentence, and induce it from the source language. Furthermore, we expect the number of verbs to be well concentrated, and we induce its maximal variance from the source language.

[6] In Section 5 we shall see that this makes optimization easier.

approach inspired by the posterior regularization method (Ganchev et al., 2010) and use the objective:

$$F_{\mathsf{c}}(A) = \min_{r(c_1,c_2)\in\mathcal{S}_{\text{pre}}} D_{KL}[r(c_1,c_2)|p_T(c_1,c_2;A)] \quad (4)$$

The above objective will attain lower values for A such that $p_T(c_1, c_2; A)$ is close to the constraint set. Specifically, it will have a value of zero when the bigram distribution induced by A has the property specified in $\mathcal{S}_{\text{pre}}$. We similarly define a set $\mathcal{S}_{\text{post}}$ for post-positional languages.

As a second typological feature, we consider the Demonstrative-Noun ordering. In DN languages we want the probability of a determiner to come before C_2 = {noun, adj, num}, (i.e., frequent universal noun-phrase tags), to cross a threshold. This constraint translates to:

$$\sum_{c\in C_2} p_T(\text{det}, c) \geq a_{\text{det}} \quad (5)$$

where $a_{\text{det}} = \sum_{c\in C_2} p_S(\text{det}, c)$ is a threshold determined from the source language. We denote the set of distributions that have this property by $\mathcal{S}_{\text{DN}}$, and add them to the constraint in (4). The overall constraint set is denoted by $\mathcal{S}$.

4.2 The Overall Objective

We have defined a set of functions $F_i(A)$ that are expected to have low values for good mappings. To combine those, we use a weighted sum: $F_\alpha(A) = \sum_i \alpha_i \cdot F_i(A)$. (The weights in this equation are learned; we discussed the procedure in Section 5)

Optimizing over the set of mappings is difficult since each mapping is a discrete set whose size is exponential size in L_T. Technically, the difficulty comes from the requirement that elements of A are integral and its columns sum to one. To relax this restriction, we will allow $A(c|f) \in [0, 1]$ and encourage A to correspond to a mapping by adding an entropy regularization term:

$$H[A] = -\sum_f \sum_c A(c|f) \log A(c|f) \quad (6)$$

This term receives its minimal value when the conditional probability of the universal tags given a language-specific tag is 1 for one universal tag and zero for the others.

The overall objective is then: $F(A) = F_\alpha(A) + \lambda \cdot H[A]$, where λ is the weight of the entropy term.[7] The resulting optimization problem is:

$$\min_{A\in\Delta} F(A) \quad (7)$$

where Δ is the set of non-negative matrices whose columns sum to one:

$$\Delta = \left\{ A : \begin{array}{ll} A(c|f) \geq 0 & \forall c, f \\ \sum_{c=1}^{K} A(c|f) = 1 & \forall f \end{array} \right\} \quad (8)$$

5 Parameter Estimation

In this section we describe the parameter estimation process for our model. We start by describing how to optimize A. Next, we discuss the weight selection algorithm, and finally the method for choosing source languages.

5.1 Optimizing the Mapping A

Recall that our goal is to solve the optimization problem in Eq. (7). This objective is non convex since the function $H[A]$ is concave, and the objective $F(A)$ involves bilinear terms in A and logarithms of their sums (see Equations (1) and (2)).

While we do not attempt to solve the problem globally, we do have a simple update scheme that monotonically decreases the objective. The update can be derived in a similar manner to expectation maximization (EM) (Neal and Hinton, 1999) and convex concave procedures (Yuille and Rangarajan, 2003). Figure 2 describes our optimization algorithm. The key ideas in deriving it are using posterior distributions as in EM, and using a variational formulation of entropy. The term $F_c(A)$ is handled in a similar way to the posterior regularization algorithm derivation. A detailed derivation is provided in the supplementary file. [8]

The k^{th} iteration of the algorithm involves several steps:

- In step 1, we calculate the current estimate of the bigram distribution over tags, $p_T(c_1, c_2; A^k)$.

[7] Note that as $\lambda \to \infty$, only valid maps will be selected by the objective.

[8] The supplementary file is available at `http://groups.csail.mit.edu/rbg/code/unitag/emnlp2012`.

- In step 2, we find the bigram distribution in the constraint set $\mathcal{S}$ that is closest in KL divergence to $p_T(c_1, c_2; A^k)$, and denote it by $r^k(c_1, c_2)$. This optimization problem is convex in $r(c_1, c_2)$.

- In step 3, we calculate the bigram posterior over language specific tags given a pair of universal tags. This is analogous to the standard E-step in EM.

- In step 4, we use the posterior in step 3 and the bigram distributions $p_S(c_1, c_2)$ and $r^k(c_1, c_2)$ to obtain joint counts over language specific and universal bigrams.

- In step 5, we use the joint counts from step 4 to obtain counts over pairs of language specific and universal tags.

- In step 6, analogous to the M-step in EM, we optimize over the mapping matrix A. The objective is similar to the Q function in EM, and also includes the $F_{verb}(A)$ term, and a linear upper bound on the entropy term. The objective can be seen to be convex in A.

As mentioned above, each of the optimization problems in steps 2 and 6 is convex, and can therefore be solved using standard convex optimization solvers. Here, we use the CVX package (Grant and Boyd, 2008; Grant and Boyd, 2011). It can be shown that the algorithm improves $F(A)$ at every iteration and converges to a local optimum.

The above algorithm generates a mapping A that may contain fractional entries. To turn it into a *hard* mapping we round A by mapping each f to the c that maximizes $A(c|f)$ and then perform greedy improvement steps (one f at a time) to further improve the objective. The regularization constant λ is tuned to minimize the $F_\alpha(A)$ value of the rounded A.

5.2 Learning the Objective Weights

Our $F_\alpha(A)$ objective is a weighted sum of the individual $F_i(A)$ functions. In the following, we describe how to learn the α_i weights for every target language. We would like $F_\alpha(A)$ to have low values when A is a *good* map. Since our performance goal is parsing accuracy, we consider a map to be good

Initialize A^0.
Repeat

Step 1 (calculate current bigram estimate):

$$p_T(c_1, c_2; A^k) = \sum_{f_1, f_2} A^k(c_1|f_1) A^k(c_2|f_2) p_T(f_1, f_2)$$

Step 2 (incorporate constraints):

$$r^k(c_1, c_2) = \arg\min_{r \in \mathcal{S}} D_{KL}[r(c_1, c_2) | p_T(c_1, c_2; A^k)]$$

Step 3 (calculate model posterior):

$$p(f_1, f_2 | c_1, c_2; A^k) \propto A^k(c_1|f_1) A^k(c_2|f_2) p_T(f_1, f_2)$$

Step 4: (complete joint counts):

$$N^k(c_1, c_2, f_1, f_2) = p(f_1, f_2 | c_1, c_2; A^k) \left(r^k(c_1, c_2) + p_S(c_1, c_2) \right)$$

Step 5 (obtain pairwise):

$$M^k(c, f) = N_1^k(c, f) + N_2^k(c, f)$$

where $N_1^k(c, f) = \sum_{c_2, f_2} N^k(c, c_2, f, f_2)$ and similarly for $N_2^k(c, f)$.

Step 6 (M step with entropy linearization): Set A^{k+1} to be the solution of

$$\min_{A \in \Delta} - \sum_{c, f} \left[M^k(c, f) \log A(c|f) + A(c|f) \log A^k(c|f) \right] + F_{verb}(A)$$

Until Convergence of A^k

Figure 2: An iterative algorithm for minimizing our objective in Eq. (7). For simplicity we assume that all the weights α_i and λ are equal to one. It can be shown that the objective monotonically decreases in every iteration.

if it results in high parsing accuracy, as measured when projecting a parser from to S to T.

Since we do not have annotated parses in T, we use the other source languages $S = \{S_1, \ldots, S_n\}$ to learn the weight. For each S_i as the target, we first train a parser for each language in $S \setminus \{S_i\}$ as if it was the source, using the map of Petrov et al. (2011), and choose $S_i^* \in S \setminus \{S_i\}$ which gives the highest parsing accuracy on S_i. Next we generate 7000 candidate mappings for S_i by randomly perturbing the map of (Petrov et al., 2011). We evaluate the quality of each candidate A by projecting the parser of S_i^* to S_i, and recording the parsing accuracy. Among all the candidates we choose the highest accuracy one and denote it by $A^*(S_i)$. We now want the score $F(A^*(S_i))$ to be lower than that of all other candidates. To achieve this, we train a ranking SVM whose inputs are pairs of maps $A^*(S_i)$ and an-

other *worse* $A(S_i)$. These map pairs are taken from many different traget languages, i.e. many different S_i. The features given to the SVM are the terms of the score $F_i(A)$. The goal of the SVM is to weight these terms such that the better map $A^*(S_i)$ has a lower score. The weights assigned by the SVM are taken as α_i.

5.3 Source Language Selection

As noted in Section 4 we construct $F(A)$ by choosing a single source language S. Here we describe the method for choosing S. Our goal is to choose S that is closest to T in terms of typology. Assume that languages are described by binary typological vectors $\mathbf{v}_L$. We would like to learn a diagonal matrix D such that $d(S,T;D) = (\mathbf{v}_S - \mathbf{v}_T)^T D(\mathbf{v}_S - \mathbf{v}_T)$ reflects the similarity between the languages. In our context, a good measure of similarity is the performance of a parser trained on S and projected on T (using the optimal map A). We thus seek a matrix D such that $d(S,T;D)$ is ranked according to the parsing accuracy. The matrix D is trained using an SVM ranking algorithm that tries to follow the ranking of parsing accuracy. Similar to the technique for learning the objective weights, we train across many pairs of source languages.[9]

The typological features we use are a subset of the features described in "The World Atlas of Languages Structure" (WALS, (Haspelmath et al., 2005)), and are shown in Table 1.

6 Evaluation Set-Up

Datasets We test our model on 19 languages: Arabic, Basque, Bulgarian, Catalan, Chinese, Czech, Danish, Dutch, English, German, Greek, Hungarian, Italian, Japanese, Portuguese, Slovene, Spanish, Swedish, and Turkish. Our data is taken from the CoNLL 2006 and 2007 shared tasks (Buchholz and Marsi, 2006; Nivre et al., 2007). The CoNLL datasets consist of manually created dependency trees and language-specific POS tags. Following Petrov et al. (2011), our model maps these language-specific tags to a set of 12 universal tags: noun, verb, adjective, adverb, pronoun, determiner, adposition, numeral, conjunction, particle, punctuation mark and X (a general tag).

[9] Ties are broken using the $F(A)$ objective.

Evaluation Procedure We perform a separate experiment for each of the 19 languages as the target and a source language chosen from the rest (using the method from Section 5.3). For the selected source language, we assume access to the mapping of Petrov et al. (2011).

Evaluation Measures We evaluate the quality of the derived mapping in the context of the target language parsing accuracy. In both the training and test data, the language-specific tags are replaced with universal tags: Petrov's tags for the source languages and learned tags for the target language. We train two non-lexicalized parsers using source annotations and apply them to the target language. The first parser is a non-lexicalized version of the MST parser (McDonald et al., 2005) successfully used in the multilingual context (McDonald et al., 2011). In the second parser, parameters of the target language are estimated as a weighted mixture of parameters learned from supervised source languages (Cohen et al., 2011). For the parser of Cohen et al. (2011), we trained the model on the four languages used in the original paper — English, German, Czech and Italian. When measuring the performance on each of these four languages, we selected another set of four languages with a similar level of diversity.[10]

Following the standard evaluation practice in parsing, we use directed dependency accuracy as our measure of performance.

Baselines We compare mappings induced by our model against three baselines: the manually constructed mapping of Petrov et al. (2011), a randomly constructed mapping and a greedy mapping. The greedy mapping uses the same objective as our full model, but optimizes it using a greedy method. In each iteration, this method makes $|L_T|$ passes over the language-specific tags, selecting a substitution that contributes the most to the objective.

Initialization To reduce the dimension of our algorithm's search space and speed up our method, we start by clustering the language-specific POS tags of the target into $|K| = 12$ clusters using an unsuper-

[10] We also experimented with a version of the Cohen et al. (2011) model trained on all the source languages. This set-up resulted in decreased performance. For this reason, we chose to train the model on the four languages.

ID	Feature Description	Values
81A	Order of Subject, Object and Verb	SVO, SOV, VSO, VOS, OVS, OSV
85A	Order of Adposition and Noun	Postpositions, Prepositions, Inpositions
86A	Order of Genitive and Noun	Genitive-Noun, Noun-Genitive
87A	Order of Adjective and Noun	Adjective-Noun, Noun-Adjective
88A	Order of Demonstrative and Noun	Demonstrative-Noun, Noun-Demonstrative, before and after

Table 1: The set of typological features that we use for source language selection. The first column gives the ID of the feature as listed in WALS. The second column describes the feature and the last column enumerates the allowable values for each feature; besides these values each feature can also have a value of 'No dominant order'.

vised POS induction algorithm (Lee et al., 2010).[11] Our mapping algorithm then learns the connection between these clusters and universal tags.

For initialization, we perform multiple random restarts and select the one with the lowest final objective score.

7 Results

We first present the results of our model using the gold POS tags for the target language. Table 2 summarizes the performance of our model and the baselines.

Comparison against Baselines On average, the mapping produced by our model yields parsers with higher accuracy than all of the baselines. These results are consistent for both parsers (McDonald et al., 2011; Cohen et al., 2011). As expected, random mappings yield abysmal results — 20.2% and 12.7% for the two parsers. The low accuracy of parsers that rely on the *Greedy* mapping — 29.9% and 25.4% — show that a greedy approach is a poor strategy for mapping optimization.

Surprisingly, our model slightly outperforms the mapping of (Petrov et al., 2011), yielding an average accuracy of 56.7% as compared to the 55.4% achieved by its manually constructed counterpart for the direct transfer method (McDonald et al., 2011). Similar results are observed for the mixture weights parser (Cohen et al., 2011). The main reason for these differences comes from mistakes introduced in the manual mapping. For example, in Czech tag "R" is labeled as "pronoun", while actually it should be mapped to "adposition". By correcting this mistake, we gain 5% in parsing accuracy for the direct transfer parser.

[11] This pre-clustering results in about 3% improvement, presumably since it uses contextual information beyond what our algorithm does.

Overall, the manually constructed mapping and our model's output disagree on 21% of the assignments (measured on the token level). However, the extent of disagreement is not necessarily predictive of the difference in parsing performance. For instance, the manual and automatic mappings for Catalan disagree on 8% of the tags and their parsing accuracy differs by 5%. For Greek on the other hand, the disagreement between mappings is much higher — 17%, yet the parsing accuracy is very close. This phenomenon shows that not all mistakes have equal weight. For instance, a confusion between "pronoun" and "noun" is less severe in the parsing context than a confusion between "pronoun" and "adverb".

Impact of Language Selection To assess the quality of our language selection method, we compare the model against an oracle that selects the best source for a given target language. As Table 2 shows our method is very close to the oracle performance, with only 0.7% gap between the two. In fact, for 10 languages our method correctly predicts the best pairing. This result is encouraging in other contexts as well. Specifically, McDonald et al. (2011) have demonstrated that projecting from a single oracle-chosen language can lead to good parsing performance, and our technique may allow such projection without an oracle.

Relations between Objective Values and Optimization Performance The suboptimal performance of the Greedy method shows that choosing a good optimization strategy plays a critical role in finding the desired mapping. A natural question to ask is whether the objective value is predictive of the end goal parsing performance. Figure 3 shows the objective values for the mappings computed by our method and the baselines for four languages. Over-

	Direct Transfer Parser (Accuracy)					Mixture Weight Parser (Accuracy)				Tag Diff.
	Random	Greedy	Petrov	Model	Best Pair	Random	Greedy	Petrov	Model.	
Catalan	15.9	32.5	74.8	**79.3**	79.3	12.6	24.6	65.6	**73.9**	8.8
Italian	16.4	41.0	**68.7**	68.3	71.4	11.7	33.5	**64.2**	61.9	6.7
Portuguese	15.8	24.6	72.0	**75.1**	75.1	10.7	14.1	70.4	**72.6**	12.2
Spanish	11.5	27.4	**72.1**	68.9	68.9	6.4	26.5	58.8	**62.8**	7.5
Danish	35.5	23.7	**46.6**	46.5	49.2	4.2	23.7	51.4	**51.7**	5.0
Dutch	18.0	22.1	**58.2**	56.8	57.3	7.1	15.3	**54.9**	53.2	4.9
English	14.7	19.0	**51.6**	49.0	49.0	13.3	15.1	**47.5**	41.8	17.7
German	15.8	24.3	**55.7**	50.4	51.6	20.9	18.7	**52.4**	51.8	15.0
Swedish	15.1	26.3	63.1	**63.1**	63.1	9.1	36.5	55.7	**55.9**	8.2
Bulgarian	17.4	28.0	51.6	**63.4**	63.4	22.6	39.9	**64.6**	60.4	35.7
Czech	19.0	34.4	47.7	**57.3**	57.3	12.7	26.2	48.3	**55.7**	28.5
Slovene	15.6	21.8	43.5	**51.4**	52.8	11.3	20.7	42.2	**53.0**	38.8
Greek	17.3	19.5	**62.3**	59.7	59.8	22.0	15.2	56.2	**57.0**	17.0
Hungarian	28.4	44.1	**53.8**	52.3	52.3	4.0	43.8	46.4	**51.7**	18.1
Arabic	22.1	45.4	**51.5**	51.2	52.9	3.9	40.9	48.3	**51.1**	15.7
Basque	18.0	19.2	27.9	**33.1**	35.1	6.3	8.3	**32.3**	30.6	43.8
Chinese	22.4	34.1	46.0	**47.6**	49.5	17.7	34.9	**44.0**	40.4	38.1
Japanese	36.5	46.2	51.4	**53.6**	53.6	15.4	18.0	25.7	**28.7**	73.8
Turkish	28.8	34.9	**53.2**	49.8	49.8	19.7	20.3	**27.7**	27.5	9.9
Average	20.2	29.9	55.4	**56.7**	57.4	12.7	25.4	50.8	**51.7**	21.3

Table 2: Directed dependency accuracy of our model and the baselines using gold POS tags for the target language. The first section of the table is for the direct transfer of the MST parser (McDonald et al., 2011). The second section is for the weighted mixture parsing model (Cohen et al., 2011). The first two columns (Random and Greedy) of each section present the parsing performance with a random or a greedy mapping. The third column (Petrov) shows the results when the mapping of Petrov et al. (2011) is used. The fourth column (Model) shows the results when our mapping is used and the fifth column in the first section (Best Pair) shows the performance of our model when the best source language is selected for every target language. The last column (Tag Diff.) presents the difference between our mapping and the mapping of Petrov et al. (2011) by showing the percentage of target language tokens for which the two mappings select a different universal tag.

all, our method and the manual mappings reach similar values, both considerably better than other baselines. While the parsing performance correlates with the objective, the correlation is not perfect. For instance, on Greek our mapping has a better objective value, but lower parsing performance.

Ablation Analysis We next analyze the contribution of each component of our objective to the resulting performance.[12] The strongest factor in our objective is the distributional features capturing global statistics. Using these features alone achieves an average accuracy of 51.1%, only 5.6% less than the full model score. Adding just the verb-related constraints to the distributional similarity objectives improves the average model performance by 2.1%. Adding just the typological constraints yields a very modest performance gain of 0.5%. This is not surprising — the source language is selected to be typologically similar to the target language, and thus its distributional properties are consistent with typological features. However, adding both the verb-related constraints and the typological constraints results in a synergistic performance gain of 5.6% over the distributional similarity objective, a gain which is much better than the sum of the two individual gains.

Application to Automatically Induced POS Tags A potential benefit of the proposed method is to relate automatically induced clusters in the target language to universal tags. In our experiments, we induce such clusters using Brown clustering,[13] which

[12]The results are consistent for both parsers, here we report the accuracy for the direct transfer method (McDonald et al., 2011).

[13]In our experiments, we employ Liang's implementation http://cs.stanford.edu/∼pliang/software/. The number of clusters is set to 30.

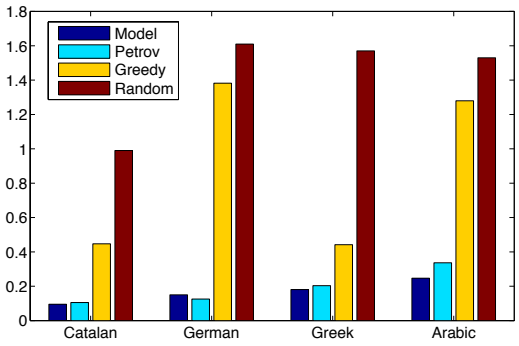


Figure 3: Objective values for the different mappings used in our experiments for four languages. Note that the goal of the optimization procedure is to minimize the objective value.

has been successfully used for similar purposes in parsing research (Koo et al., 2008). We then map these clusters to the universal tags using our algorithm.

The average parsing accuracy on the 19 languages is 45.5%. Not surprisingly, automatically induced tags negatively impact parsing performance, yielding a decrease of 11% when compared to mappings obtained using manual POS annotations (see Table 2). To further investigate the impact of inaccurate tags on the mapping performance, we compare our model against the oracle mapping model that maps each cluster to the most common universal tag of its members. Parsing accuracy obtained using this method is 45.1%, closely matching the performance of our mapping algorithm.

An alternative approach to mapping words into universal tags is to directly partition words into K clusters (without passing through language specific tags). In order for these clusters to be meaningful as universal tags, we can provide several prototypes for each cluster (e.g., "walk" is a verb etc.). To test this approach we used the prototype driven tagger of Haghighi and Klein (2006) with 15 prototypes per universal tag.[14] The resulting universal tags yield an average parsing accuracy of 40.5%. Our method (using Brown clustering as above) outperforms this baseline by about 5%.

[14]Oracle prototypes were obtained by taking the 15 most frequent words for each universal tag. This yields almost the same total number of prototypes as those in the experiment of (Haghighi and Klein, 2006).

8 Conclusions

We present an automatic method for mapping *language-specific* part-of-speech tags to a set of *universal tags*. Our work capitalizes on manually designed conversion schemes to automatically create mappings for new languages. Our experimental results demonstrate that automatically induced mappings rival the quality of their hand-crafted counterparts. We also establish that the mapping quality has a significant impact on the accuracy of syntactic transfer, which motivates further study of this topic. Finally, our experiments show that the choice of mapping optimization scheme plays a crucial role in the quality of the derived mapping, highlighting the importance of optimization for the mapping task.

Acknowledgments

The authors acknowledge the support of the NSF (IIS-0835445), the MURI program (W911NF-10-1-0533) and the DARPA BOLT program. We thank Tommi Jaakkola, the members of the MIT NLP group and the ACL reviewers for their suggestions and comments. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the authors, and do not necessarily reflect the views of the funding organizations.

References

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of CoNLL*, pages 149–164.

David Burkett and Dan Klein. 2008. Two languages are better than one (for syntactic parsing). In *Proceedings of EMNLP*, pages 877–886.

Shay B. Cohen, Dipanjan Das, and Noah A. Smith. 2011. Unsupervised structure prediction with non-parallel multilingual guidance. In *EMNLP*, pages 50–61.

Frank Van Eynde. 2004. Part of speech tagging en lemmatisering van het corpus gesproken nederlands. In *Technical report*.

Jenny Rose Finkel, Trond Grenager, and Christopher D. Manning. 2007. The infinite tree. In *ACL*.

Kuzman Ganchev, Joao Graca, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *JMLR*.

Michael R. Garey and David S. Johnson. 1979. *Computers and Intractability: A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York, NY, USA.

M. Grant and S. Boyd. 2008. Graph implementations for nonsmooth convex programs. In V. Blondel, S. Boyd, and H. Kimura, editors, *Recent Advances in Learning and Control*, Lecture Notes in Control and Information Sciences, pages 95–110. Springer-Verlag Limited.

M. Grant and S. Boyd. 2011. CVX: Matlab software for disciplined convex programming, version 1.21, April.

Aria Haghighi and Dan Klein. 2006. Prototype-driven learning for sequence models. In *HLT-NAACL*.

Martin Haspelmath, Matthew S. Dryer, David Gil, and Bernard Comrie, editors. 2005. *The World Atlas of Language Structures*. Oxford University Press.

R. Hwa, P. Resnik, A. Weinberg, C. Cabezas, and O. Kolak. 2005. Bootstrapping parsers via syntactic projection across parallel texts. *Journal of Natural Language Engineering*, 11(3):311–325.

Dan Klein and Christopher Manning. 2003. Accurate unlexicalized parsing. In *ACL*.

Terry Koo, Xavier Carreras, and Michael Collins. 2008. Simple semi-supervised dependency parsing. In *Proceedings of the ACL*, pages 595–603.

Yoong Keok Lee, Aria Haghighi, and Regina Barzilay. 2010. Simple type-level unsupervised pos tagging. In *EMNLP*.

Percy Liang, Slav Petrov, Michael I. Jordan, and Dan Klein. 2007. The infinite pcfg using hierarchical dirichlet processes. In *EMNLP-CoNLL*.

Takuya Matsuzaki, Yusuke Miyao, and Jun'ichi Tsujii. 2005. Probabilistic cfg with latent annotations. In *ACL*.

Ryan McDonald, Fernando Pereira, Kiril Ribarov, and Jan Hajic. 2005. Non-projective dependency parsing using spanning tree algorithms. In *HLT-EMNLP*.

Ryan T. McDonald, Slav Petrov, and Keith Hall. 2011. Multi-source transfer of delexicalized dependency parsers. In *EMNLP*, pages 62–72.

Tahira Naseem, Harr Chen, Regina Barzilay, and Mark Johnson. 2010. Using universal linguistic knowledge to guide grammar induction. In *EMNLP*, pages 1234–1244.

Radford M. Neal and Geoffrey E. Hinton. 1999. A view of the em algorithm that justifies incremental, sparse, and other variants. In Michael I. Jordan, editor, *Learning in graphical models*, pages 355–368. MIT Press, Cambridge, MA, USA.

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.

Slav Petrov and Dan Klein. 2007. Improved inference for unlexicalized parsing. In *NAACL-HLT*.

Slav Petrov, Leon Barrett, Romain Thibaux, and Dan Klein. 2006. Learning accurate, compact, and interpretable tree annotation. In *ACL-COLING*.

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

Chenhai Xi and Rebecca Hwa. 2005. A backoff model for bootstrapping resources for non-english languages. In *Proceedings of EMNLP*, pages 851 – 858.

Alan Yuille and Anand Rangarajan. 2003. The concave-convex procedure (cccp). In *Neural Computation*.

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, January.