Part-of-speech Taggers for Low-resource Languages using CCA Features

Young-Bum Kim[†] **Benjamin Snyder**[‡] **Ruhi Sarikaya**[†]

[†]Microsoft Corporation, Redmond, WA
[‡]University of Wisconsin-Madison, Madison, WI
{ybkim, ruhi.sarikaya}@microsoft.com
bsnyder@cs.wisc.edu

Abstract

In this paper, we address the challenge of creating accurate and robust part-of-speech taggers for low-resource languages. We propose a method that leverages existing parallel data between the target language and a large set of resource-rich languages without ancillary resources such as tag dictionaries. Crucially, we use CCA to induce latent word representations that incorporate cross-genre distributional cues, as well as projected tags from a full array of resource-rich languages. We develop a probability-based confidence model to identify words with highly likely tag projections and use these words to train a multi-class SVM using the CCA features. Our method yields average performance of 85% accuracy for languages with almost no resources, outperforming a state-of-the-art partially-observed CRF model.

1 Introduction

We address the challenge of creating accurate and robust part-of-speech taggers for low-resource languages. We aim to apply our methods to the hundreds, and potentially thousands, of languages with meager electronic resources. We do not assume the existence of a tag dictionary, or any other sort of prior knowledge of the target language. Instead, we base our methods entirely on the existence of parallel data between the target language and a set of resource-rich languages.

Fortunately, such parallel data exists for just about every written language, in the form of Bible translations. Around 2,500 languages have at least partial Bible translations, and somewhere between 500 and 1,000 languages have complete translations. We have collected such electronic Bible translations for 650 languages. Figure 1 breaks down the number of languages in our collection according to their token count. The majority of our languages have at least 200,000 tokens of Bible translations.

While previous studies (Täckström et al., 2013; Ganchev and Das, 2013) have addressed this general setting, they have typically assumed the existence of a partial tag dictionary as well as large quantities of non-parallel data in the target language. These assumptions are quite reasonable for the dozen most popular languages in the world, but are inadequate for the creation of a truly world-wide repository of NLP tools and linguistic data.

In fact, we argue that such ancillary sources of information are not really necessary once we take into account the vastly multilingual nature of our parallel data. Annotations projected from individual resource-rich languages are often noisy and unreliable, due to systematic differences between the languages in question, as well as word alignment errors. We can thus think of these languages as very lazy and unreliable annotators of our target language. Despite their incompetence, as the number of such annotators increases, their combined efforts converge upon the truth, as idiosyncratic biases and random noise are washed away.

Our assumption throughout will be that we have in our possession a single multilingual corpus (the Bible) consisting of about 200,000 tokens for several hundred languages languages, as well as reasonably accurate POS taggers for about ten "resource-rich" languages. We will tag the Bible data for the resource-rich languages, word-align them to one another, and also word-align them to

Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, pages 1304–1314,
Lisbon, Portugal, 17-21 September 2015.

the remaining several hundred target languages.

Of course, our goal is not to produce a tagger restricted to the Biblical lexicon. We therefore assume a small unannotated monolingual sample of the target language in an entirely unrelated genre (e.g. newswire). We use this sample transductively to adapt our learned taggers from the Biblical genre. In our experiments, we use the CoNLL 2006 and 2007 shared-task test data for this purpose. Of course tagged data does not exist for truly resource-poor languages, so we evaluate our methodology on the resource-rich languages. Each such language takes a turn playing the role of the target language for testing purposes.

The goal of the paper is to introduce a general "recipe" for successful cross-lingual induction of accurate taggers using meager resources. We faced three major technical challenges:

- First, word alignments across languages are incomplete, and often do not preserve part-of-speech due to language differences.
- Second, when using multiple resource-rich languages, we need to resolve conflicting projections.
- Third, the parallel data at our disposal is of an idiosyncratic genre (the Bible), and we wish to induce a general-purpose tagger.

To address these challenges, we forgo the typical sequence-based learning technique of HMM's and CRF's and instead adopt an instance-learning approach using latent distributional features. To induce these features, we introduced a new method using Canonical Correlation Analysis (CCA) to generalize the aligned information to new words. This method views each word position as consisting of three fundamental views: (1) the token view (word context), (2) the type view, and (3) the projected tags in the local vicinity. We perform a CCA to induce latent continuous vector representations of each view that maximizes their correlations to one another. On the test data, a simple multi-class classifier then suffices to predict accurate tags, even for novel words. This approach outperform a state-of-the-art baseline (Täckström et al., 2013) to achieve average tag accuracy of 85% on newswire text.

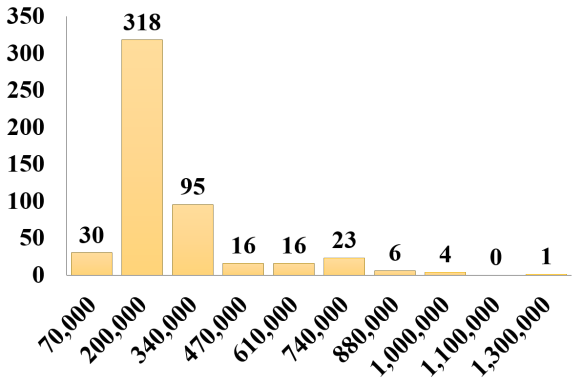

Figure 1: The breakdown of languages by the number of tokens in their available Bible translations. The horizontal axis gives the number of tokens, and the vertical axis gives the number of languages in each token range.

2 Related Work

2.1 Multilingual Projection

The idea of projecting annotated resources across languages using parallel data was first proposed by Yarowsky et al. (2001). This early work recognized the noisy nature of automatic word alignments and engineered smoothing and filtering methods to mitigate the effects of cross- lingual variation and alignment errors. More recent work in this vein has dealt with this by instead transferring information at the word type or model structure level, rather than on a token-by-token basis (Das and Petrov, 2011; Durrett et al., 2012). Current state-of-the-art results for indirectly supervised POS performance use a combination of token constraints as well as type constraints mined from Wiktionary (Li et al., 2012; Täckström et al., 2013; Ganchev and Das, 2013). As we argued above, the only widely available source of information for most low-resource languages is in fact their Bible translation. Perhaps surprisingly, our experiments show that this data source suffices to achieve state-of-the-art results.

Several previous authors have considered the advantage of using more than one resource-rich language to alleviate alignment noise.

Fossum and Abney (2005) found that using two source languages project-sources gave better results than simply using more data from one language. McDonald et al. (2011) also found advantages to using multiple language sources for projecting parsing constraints. In more of an unsu-

pervised context (but using small tag dictionaries), adding more languages to the mix has been shown to improve part-of-speech performance across all component languages (Naseem et al., 2009).

In our own previous multilingual work, we have developed the idea that supervised knowledge of some number of languages can help guide the unsupervised induction of linguistic structure, even in the absence of parallel text (Kim et al., 2011; Kim and Snyder, 2012; Kim and Snyder, 2013a; Kim and Snyder, 2013b). We have showed that cross-lingual supervised learning leads to significant performance gains over monolingual models. We point out that the previous tasks have considered as word-level structural analyses and our present case as a sentence-level analysis.

2.2 Word Alignment

Most of the papers surveyed above rely on automatic word alignments to guide the cross-lingual transfer of information. Given our desire to use highly multilingual information to improve projection accuracy, the question of word alignment performance becomes crucial. Our hypothesis is that multiple language projections are beneficial not only in weeding out random errors and idiosyncratic variations, but also in improving the linguistic consistency of the alignments themselves. Instead of simply aligning each source language to the target language in isolation, we will instead use a confidence model to synthesize information from multiple sources.

While there are not many well-known papers that have explored word alignment on a multilingual scale[1], there have been related efforts to *symmetrize* bilingual alignment models, using a variety of techniques ranging from modifications of EM (Liang et al., 2006), posterior-regularized objective function (Ganchev et al., 2010), and by considering relaxations of the hard combinatorial assignment problem (DeNero and Macherey, 2011).

2.3 Canonical Correlation Analysis (CCA)

Our method for generalizing the projections to unseen words and contexts is based on Canonical Correlation Analysis (CCA), a dimensionality reduction technique first introduced by Hotelling (1936). The key idea is to consider two groups of random variables with corresponding observations and to find linear subspaces with highest correlation between the two views. This can be seen as a kind of supervised version of Principal Components Analysis (PCA), where each view is providing supervision for the other. In fact, it can be shown that CCA directly generalizes both multiple linear regression and Fisher's Latent Discriminative Analysis (LDA) (Glahn, 1968).

From a learning theory perspective, CCA is interesting in that it allows us to prove regret-based learning bounds that depend on the "intrinsic" dimensionality of the problem rather than the apparent dimensionality (Kakade and Foster, 2007). This seems especially relevant to natural language processing scenarios, where the ambient dimension is extremely large and sparse, but reductions to dense lower-dimensional spaces may preserve nearly all the relevant semantic and syntactic information. In fact, CCA has recently been adapted to learning latent word representations in an interesting way: by dividing each word position into a token view (which only sees surrounding context) and a type view (which only sees the word itself) and performing a CCA between these two views (Dhillon et al., 2012; Kim et al., 2014; Stratos et al., 2014; Stratos et al., 2015; Kim et al., 2015c). CCA is also used to induce label representations (Kim et al., 2015d) and lexicon representations (Kim et al., 2015b).

Our technique will extend this idea by additionally considering a third *projected tag* view. Crucially, it is this view which pushes the latent representations into coherent part-of-speech categories, allowing us to simply apply multi-class SVM for unseen words in our test set.

3 Tag projection from resource-rich languages

In this section, we describe two methods for incorporating transferred tags from resource-rich languages: sequence-based learning (Täckström et al., 2013; Kim et al., 2015a) and instance-based learning. In the former, the transferred tags are used to train a partially-observed CRF (PO-CRF) by maximizing the probability of a constrained lattice. In contrast, instance-based learning views each word token as an independent classification task, but uses latent distributional information gleaned from surrounding words as features.

[1] Mayer and Cysouw (2012) used multilingual word alignment to compare languages

3.1 A sequence learning example of partially observed CRF (PO-CRF)

A first-order CRF parametrized by $\theta \in \mathbb{R}^d$ defines a conditional probability of a label sequence $y = y_1 \dots y_n$ given an observation sequence $x = x_1 \dots x_n$ as follows:

$$p_\theta(y|x) = \frac{\exp(\theta^\top \Phi(x, y))}{\sum_{y' \in \mathcal{Y}(x)} \exp(\theta^\top \Phi(x, y'))}$$

where $\mathcal{Y}(x)$ is the set of all possible label sequences for x and $\Phi(x, y) \in \mathbb{R}^d$ is a global feature function that decomposes into local feature functions $\Phi(x, y) = \sum_{j=1}^{n} \phi(x, j, y_{j-1}, y_j)$ by the first-order Markovian assumption. Given fully labeled sequences $\{(x^{(i)}, y^{(i)})\}_{i=1}^{N}$, the standard training method is to find θ that maximizes the log likelihood of the label sequences under the model with l_2-regularization:

$$\theta^* = \arg\max_{\theta \in \mathbb{R}^d} \sum_{i=1}^{N} \log p_\theta(y^{(i)}|x^{(i)}) - \frac{\lambda}{2} ||\theta||^2$$

We used an l_2 penalty weight λ of 1. Unfortunately, in our setting, we do not have fully labeled sequences. Instead, for each token x_j in sequence $x_1 \dots x_n$ we have the following two sources of label information:

- A set of allowed label types $\mathcal{Y}(x_j)$. (Label dictionary, type constraints)
- Labels $\tilde{y}_j$ transferred from resource rich languages. (transferred labels, token constraints)

Following previous work of Täckström et al. (2013), we first define a constrained *lattice* $\mathcal{Y}(x, \tilde{y}) = \mathcal{Y}(x_1, \tilde{y}_1) \times \dots \times \mathcal{Y}(x_n, \tilde{y}_n)$ where at each position j a set of allowed label types is given as:

$$\mathcal{Y}(x_j, \tilde{y}_j) = \begin{cases} \{\tilde{y}_j\} & \text{if } \tilde{y}_j \text{ is given} \\ \mathcal{Y}(x_j) & \text{otherwise} \end{cases}$$

And then we can define a conditional probability over label lattices for a given observation sequence x:

$$p_\theta(\mathcal{Y}(x, \tilde{y})|x) = \sum_{y \in \mathcal{Y}(x, \tilde{y})} p_\theta(y|x)$$

Given a label dictionary $\mathcal{Y}(x_j)$ for every token type x_j and training sequences $\{(x^{(i)}, \tilde{y}^{(i)})\}_{i=1}^{N}$ where $\tilde{y}^{(i)}$ is transferred labels for $x^{(i)}$ and, the new training method is to find θ that maximizes the log likelihood of the label lattices:

$$\theta^* = \arg\max_{\theta \in \mathbb{R}^d} \sum_{i=1}^{N} \log p_\theta(\mathcal{Y}(x^{(i)}, \tilde{y}^{(i)})|x^{(i)}) - \frac{\lambda}{2} ||\theta||^2$$

Since this objective is non-convex, we find a local optimum with a gradient-based algorithm. The gradient of this objective at each example $(x^{(i)}, \tilde{y}^{(i)})$ takes an intuitive form:

$$\begin{aligned} &\frac{\partial}{\partial \theta} \log p_\theta(\mathcal{Y}(x^{(i)}, \tilde{y}^{(i)})|x^{(i)}) - \frac{\lambda}{2} ||\theta||^2 \\ &= \sum_{y \in \mathcal{Y}(x^{(i)}, \tilde{y})} p_\theta(y|x^{(i)}) \Phi(x^{(i)}, y) \\ &- \sum_{y \in \mathcal{Y}(x^{(i)})} p_\theta(y|x^{(i)}) \Phi(x^{(i)}, y) - \lambda\theta \end{aligned}$$

This is the same as the standard CRF training except the first term where the gold features $\Phi(x^{(i)}, y^{(i)})$ are replaced by the expected value of features in the constrained lattice $\mathcal{Y}(x^{(i)}, \tilde{y})$.

An important distinction in our setting is that our token and type constraints are generated by only using the transferred tags whereas Täckström et al. (2013) generate type constraints induced from Wiktionary. Our setting is more realistic for at least two reasons; 1) Wiktionary is not always available. 2) transferable information is not limited, but Wiktionary is (e.g., semantic role and named entity).

3.2 Cross-lingual instance-based learning

The proposed method for cross-lingual instance-based learning has three steps:

1. Select training tokens based on the confidence of the projected tag information.
2. Induce distributional features over these words that incorporate all projected tags.
3. Train a multi-class classifier with these induced features to make local predictions for individual tokens.

We will describe each step below.

3.2.1 Selecting training words

Since transferred tags are not always reliable, all words in the parallel data are not necessary helpful in training. Since this method trains on words

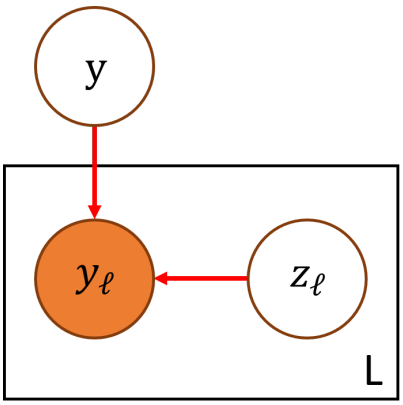

Figure 2: Graphical representation of the confidence model. Unobserved variable y denotes the true target-language tag for a token. Each of the L resource-rich languages displays a project of y, as y_ℓ, with an indicator variable z_ℓ determining the fidelity of the projection.

instead of sequences, it is easy to discard words which have unreliable or highly conflicting projections from different resource-rich languages.

To select our set of training tokens, we define a simple probability-based *confidence* model, illustrated in Figure 2. Suppose we have L resource-rich languages with alignments to the word in question. If the true tag is y, we assume that the projected tag for language ℓ will be identical to y with probability $1 - \epsilon_\ell$, where ϵ_ℓ is a language-specific corruption probability. With probability ϵ_ℓ, the projection will instead be chosen randomly (uniformly).

To make this explicit, we introduce a corruption indicator variable z_ℓ with:

$$P(z_\ell = 1) = \epsilon_\ell$$

Given z_ℓ, the probability of the projected tag y_ℓ is given by:

$$P(y_\ell|y, z_\ell) = \begin{cases} 1 & \text{if } z = 0 \text{ and } y = y_\ell, \\ \frac{1}{m} & \text{if } z = 1, \\ 0 & \text{otherwise.} \end{cases}$$

where m is the total number of possible tags. We can now compute a conditional distribution over the unknown tag y, marginalizing out the unknown corruption variables for each language:

$$p(y|y_1, \ldots, y_n) = \frac{\prod_{\ell=1}^{n} \left[\frac{\epsilon_\ell}{m} + (1 - \epsilon_\ell)\delta(y, y_\ell)\right]}{\frac{1}{m^{n-1}} \sum_{y' \in Y} \prod_{\ell=1}^{n} \left[\frac{\epsilon_\ell}{m} + (1 - \epsilon_\ell)\delta(y', y_\ell)\right]}$$

where Y is all possible tags. For simplicity, we simply set all ϵ_ℓ to 0.1 and use y as a training label when the conditional probability of the most likely value is greater than 0.9.

3.2.2 Inducing distributional features

In this section we discuss our approach for deriving latent distributional features. Canonical Correlation Analysis (CCA) is a general method for inducing new representations for a pair of variables X and Y (Hotelling, 1936). To derive word embeddings using CCA, a natural approach is to define X to represent a word and Y to represent the relevant information about a word, typically context words (Dhillon et al., 2012; Kim et al., 2015c). When they are defined as one-hot encodings, the CCA computation reduces to performing an SVD of the matrix Ω where each entry is

$$\Omega_{w,c} = \frac{\textbf{count}(w, c)}{\sqrt{\textbf{count}(w)\textbf{count}(c)}}$$

where $\textbf{count}(w, c)$ denotes co-occurrence count of word w and context c in the given corpus, $\textbf{count}(w) = \sum_c \textbf{count}(w, c)$, and $\textbf{count}(c) = \sum_w \textbf{count}(w, c)$.

The resulting word representation is given by $U^\top X$ where U is a matrix of the scaled left singular vectors of Ω (See Figure 3). In our work, we use a slightly modified version of this definition by taking square-root of each count:

$$\sqrt{\Omega}_{w,c} = \frac{\textbf{count}(w, c)^{1/2}}{\sqrt{\textbf{count}(w)^{1/2}\textbf{count}(c)^{1/2}}}$$

This has an effect of stabilizing the variance of each term in the matrix, leading to a more efficient estimator. The square-root transformation also transforms the distribution of the count data to look more Gaussian (Bartlett, 1936): since an interpretation of CCA is a latent-variable with normal distributions (Bach and Jordan, 2005), it makes the data more suitable for CCA. It has been observed in past works (e.g., Dhillon et al. (2012)) to significantly improve the quality of the resulting representations.

3.3 Feature Induction Algorithm

We now describe our algorithm for inducing latent distributional features both on the multilingual parallel corpus, as well as the monolingual, newswire test data. This algorithm is described

in detail in Figure 4. The key idea is to perform two CCA steps. The first step incorporates word-distributional information over both the multilingual corpus (the Bible) as well as the external domain monolingual corpus (CONLL data)[2]. This provides us with word representations that are general, and not overly specific to any single genre. However, it does not incorporate any projected tag information. We truncate this first SVD to the first 100 dimensions[3].

After this CCA step is performed, we then replace the words in the multilingual Bible data with their latent representations. We then perform a second CCA between these word representations and vectors representing the projected tags from all resource-rich languages. This step effectively *adapts* the first latent representation to the information contained in the tag projections. We truncate this second SVD to the first 50 dimensions.

We now have word embeddings that can be applied to any corpus, and are designed to maximize correlation both with typical surrounding word context, as well as typical projected tag context. These embeddings serve as our primary feature vectors for training the POS classifier (described in the next section). We concatenate this primary feature vector with the embeddings of the previous and subsequent words, in order to provide context-sensitive POS predictions.

3.3.1 Multi-class classifier

To train our POS tagger, we use a linear multi-class SVM (Crammer and Singer, 2002). It has a parameter $w_y \in \mathbb{R}^d$ for every tag $y \in \mathcal{T}$ and defines a linear score function $s(\mathbf{x}, j, y) := w_y^\top \Phi(\mathbf{x}, j)$. Given any sentence $\mathbf{x}$ and a position j, it predicts $\arg\max_{y \in \mathcal{T}} s(\mathbf{x}, j, y)$ as the tag of x_j. We use the implementation of Fan et al. (2008) with the default hyperparameter configurations for training.

4 Experiments

4.1 Datasets and Experimental Setup

There are more than 4,000 living languages in the world, and one of the most prevalently translated books is the Bible. We now describe the Bible dataset we collected.

[2]For context words, we use 5 words before and after the word occurrence.

[3]Embedding dimension was empirically determined by the singular values.

CCA-PROJ-SPARSE
Input: samples $(x^{(1)}, y^{(1)}) \dots (x^{(n)}, y^{(n)}) \in \{0,1\}^d \times \{0,1\}^{d'}$, dimension k
Output: projections $A \in \mathbb{R}^{d \times k}$ and $B \in \mathbb{R}^{d' \times k}$

- Calculate $B \in \mathbb{R}^{d \times d'}$, $u \in \mathbb{R}^d$, and $v \in \mathbb{R}^{d'}$:

$$B_{i,j} = \sum_{l=1}^{n} [[x_i^{(l)} = 1]][[y_j^{(l)} = 1]]$$

$$u_i = \sum_{l=1}^{n} [[x_i^{(l)} = 1]] \quad v_i = \sum_{l=1}^{n} [[y_i^{(l)} = 1]]$$

- Define $\hat{\Omega} = \text{diag}(u)^{-1/2} B \text{diag}(v)^{-1/2}$.
- Calculate rank-k SVD $\hat{\Omega}$. Let $U \in \mathbb{R}^{d \times k}$ ($V \in \mathbb{R}^{d' \times k}$) be a matrix of the left (right) singular vector corresponding to the largest k singular values.
- Let $A = \text{diag}(u)^{-1/2} U$ and $B = \text{diag}(v)^{-1/2} V$.

Figure 3: Algorithm for deriving CCA projections from samples of two variables.

Lang	Tagger	Accuracy
BG	Treetagger	0.9909
CS	Hunpos	0.8969
DA	Hunpos	0.9756
DE	Hunpos	0.9855
EN	Hunpos	0.9854
ES	Treetagger	0.8785
IT	Treetagger	0.9059
NL	Treetagger	0.8781
PT	Hunpos	0.9770
AVG	-	0.9415

Table 1: Tagger accuracy on CoNLL data.

We first collect 893 bible volumes spanning several hundred languages that are freely available from three resources (www.bible.is, www.crosswire.org, www.biblegateway.com) and changed to UTF-8 format. The distribution of token in each bible in the unit of a language is in Figure 1.

Note that the Bible scripts are not exactly translated by sentences but by verses. We thus assume that each verse in a chapter has the same meaning if the number of verses is exactly same in a same chapter. We also assume that the whole chapters have the same meaning if the number of chapters in a book are exactly the same. In the same manner, we also assume the volumes that have the same number of chapters are the same. That is, their volume size should be as similar as possible

Input:

- N "labeled" tokens in the Bible domain: word $w^{(i)} \in \mathcal{V}$, corresponding context $C(w^{(i)}) \subset \mathcal{V}$ and (projected) tag set $P^{(i)} \subset \mathcal{T}$ for $i = 1 \dots N$
- N' tokens in data in the test domain: word $v^{(i)} \in \mathcal{V}'$ and corresponding context $C(v^{(i)}) \subset \mathcal{V}'$ for $i = 1 \dots N'$
- CCA dimensions k_1, k_2

Output: embedding $\mathbf{e}(w) \in \mathbb{R}^{k_2}$ for each word $w \in \mathcal{V} \cup \mathcal{V}'$

1. Combine the observed tokens and their context from the Bible and data in the test domain:

$$\mathcal{W}_1 := \left(w : w \in (w^{(i)})_{i=1}^{N} \cup (v^{(i)})_{i=1}^{N'}\right)$$
$$\mathcal{C}_1 := \left(C(w) : w \in (w^{(i)})_{i=1}^{N} \cup (v^{(i)})_{i=1}^{N'}\right)$$

2. Perform rank-k_1 **CCA-PROJ-SPARSE** on $(\mathcal{W}_1, \mathcal{C}_1)$ to derive a word projection matrix $\Phi_{\mathcal{W}_1}$ and a context projection matrix $\Phi_{\mathcal{C}_1}$.

3. Project all word examples in the Bible domain using $\Phi_{\mathcal{W}_1}$. Denote these projected words and the corresponding projected tag sets from all resource-rich languages by

$$\mathcal{W}_2 := \left(\Phi_{\mathcal{W}_1}(w^{(i)}) : i = 1 \dots N\right)$$
$$\mathcal{P}_2 := \left(P^{(i)} : i = 1 \dots N\right)$$

4. Perform rank-k_2 CCA on $(\mathcal{W}_2, \mathcal{P}_2)$ to derive a word projection matrix $\Phi_{\mathcal{W}_2}$ and a tag projection matrix $\Phi_{\mathcal{P}_2}$.

5. Set the embedding $\mathbf{e}(w)$ for each word $w \in \mathcal{V} \cup \mathcal{V}'$ as

$$\mathbf{e}(w) = \Phi_{\mathcal{W}_2}(\Phi_{\mathcal{W}_1}(w))$$

Figure 4: Algorithm for deriving word vectors for the (unannotated) test data that use the projected tags in the Bible data.

with the respect to the number of verses, chapters, and books.

Based upon these assumptions, we choose the best translation in a language based on a comparison to a reference Bible, the Modern King James Version (MKJV) in English. We choose the translation for each language that best matches this reference version in terms of chapter and verse numbering.

There are other factors considered if there are more than one candidates satisfying this matching. We focus on the contents of the bible such as the publication time. For instance, 1599 Geneva Bible in English contains old vocabulary with different spelling systems, causing unexpected errors when tagged by POS annotation tools. Also, some of volumes such as Amplified Bible (AMP) contains extraneous comments on verses themselves, causing errors for word alignments.

After the choice of the best volume, we finally select the 10 resource rich languages[4]. The two criteria to select resource rich languages are having i) the matched bible scripts both on the Old and New testament and ii) reliable parts-of-speech annotation tools. If these two requirements are satisfied, we can freely add more languages as resource rich languages in the future research. We use Hunpos tagger for CS, DA, DE, EN, and PT, Treetagger for BG, ES, IT, and NL, and Meltparser for FR.

4.2 Test Data

We use CoNLL parts-of-speech tagged data (selected resource-rich languages), plus Basque (EU), Hungarian (HU) and Turkish (TR)) as our test data. It consists of 5,000-6,000 hand-labeled tokens. The accuracy of each supervised tagger on this data is about 94% on average. Since there is no French tagged CoNLL data, we exclude French on testing but still use it in Training. The accuracy of each supervised tagger on this data is shown in Table 1.

The tag definitions used in CoNLL data are not exactly matched the ones used in the taggers when converted to universal POS tags. For instance in Spanish, we initially follow mapping of Petrov et al. (2011) for CoNLL data. The 'dp' tag for words *sus, su, mi* are mapped to DET but they are mapped to PRON in the bible data because of the Treetagger definitions. Whenever we find this kind of issues, we analyze them and choose the one of mappings for compatibility. For the 'dp' tag, we choose to map PRON.

4.3 Alignments

We perform two kinds of alignments in our data sets; (i) the verse alignment and (ii) the word alignment. When the tagged bible volumes are prepared, we align verses across all resource rich languages. For verse alignments, we pre-process to remove extraneous information such as inline reference (e.g. [REV 4:16]) and HTML tags. These alignments between two languages

[4]Bulgarian (BG), Czech(CS), Danish (DA), English(EN), German (DE), French (FR), Spanish (ES), Italian (IT), Dutch (NL), and Portuguese (PT)

occurred only when volumes have the exact same number of chapters and verses. For instance, Mark must have 16 chapters and the first chapter of the Mark must have 45 verses in our criteria. The correct number of chapters and verses are pre-defined on MKJV volume, and the number of matched verses on each volume is greater than 30,500.

After performing verse alignments, we then perform word alignments. The quality of tags in resource poor languages is highly dependent on the quality of word alignments because parts-of-speech tags will be projected through this alignment path. First, we use GIZA++ for initial one-to-many alignments and we symmetrize by taking their intersection. This ensures that the resulting alignments are of high quality.

4.4 Results

	majority	union	confident
BG	0.8123	0.8167	0.8235
CS	0.8013	0.8094	0.8142
DA	0.8412	0.8497	0.8492
DE	0.8532	0.8611	0.8721
ES	0.8278	0.8345	0.8385
EU	0.8326	0.8413	0.8472
HU	0.7741	0.7789	0.7953
IT	0.8486	0.8445	0.8481
NL	0.7864	0.7876	0.7884
PT	0.8022	0.8081	0.8110
TR	0.6803	0.6739	0.6935
AVG	0.8055	0.8097	0.8165

Table 2: Baseline model CONLL performance depending on criterion for selecting tag projection.

In all experiments, we hold out the tags of the test language. EU, HU and TR used projected tags from 10 resource-rich languages, 9 resource-rich languages are used for the remaining languages. In our first experiment, we consider the state-of-the-art PO-CRF baseline. This model trains a partially observed CRF based on a single projected tag for each token. We experiment with different methods of choosing the projected tags. The results are shown in Table 2. The majority method is to choose the most common tag from the projected tags of the current token. We then experiment with taking the union of all projected tags (i.e. only constraining the lattice based on unanimity of the resource-rich languages). Finally, we considered choosing the high confidence tags, based on our confidence model. The confident tags are defined by a method described in Section 3.2.1 If this ratio is greater than 0.9, we assume that this token has high confidence. As the results indicate, this final method yielded the best tagging performance on the CONLL test data, achieving average accuracy of 82%.

In the remaining experiments we will adopt the confidence-based selection criterion for both the baseline as well as our method.

	PO-CRF	CCA+SVM
BG	0.8450	0.8686
CS	0.8359	0.8442
DA	0.8727	0.8826
DE	0.8862	0.9025
ES	0.8523	0.8816
EU	0.8506	0.8927
HU	0.8461	0.8495
IT	0.8705	0.8911
NL	0.8115	0.8345
PT	0.8346	0.8410
TR	0.7064	0.7389
AVG	0.8375	0.8570

Table 3: Performance on multilingual Bible data

In order to isolate the errors due to projection mismatch versus domain variation, we first test both models on the Bible data itself. To do so, we assume that the tags produced by the test-language's supervised tagger are in fact the ground truth. This experiment allows us to compare to tag projection models using (1) PO-CRF and (2) CCA+SVM. Results are given in Table 3. Unsurprisingly, PO-CRF performs better on the multilingual corpus than on the CONLL data, due to the beneficial constraint of the projected tags. Perhaps interestingly, the CCA+SVM method, which is a simple instance-based classifier using cleverly constructed features, outperforms the sequence labeller, achieving accuracy of nearly 86%[5] .

In our third experiment we use CoNLL test data and compare the PO-CRF models with different settings. See Table 4. This experiment is to show the effects of suffix and Brown cluster features on PO-CRF to relieve the unseen words issue. We also show that the more projecting languages are

[5]Note that some previous researches (Liang et al., 2008; Wisniewski et al., 2014; Moore, 2014) also pointed out that POS tagging does not necessarily require a sequence model for strong performance.

	1 lang (EN) (A)	9/10 langs (W)	9/10 langs (no S/C)	9/10 langs (A)
BG	0.7883	0.7144	0.8094	0.8478
CS	0.6601	0.5589	0.6535	0.7868
DA	0.7820	0.7765	0.8016	0.8227
DE	0.8323	0.6956	0.7589	0.8500
ES	0.7893	0.7608	0.8279	0.8665
EU	0.7764	0.7543	0.8035	0.8661
HU	0.6429	0.6378	0.7834	0.8119
IT	0.8444	0.7588	0.8136	0.8921
NL	0.7887	0.6825	0.7751	0.8214
PT	0.8476	0.7797	0.8464	0.8656
TR	0.6306	0.5727	0.6719	0.7143
AVG	0.7621	0.6993	0.7768	0.8314

Table 4: Accuracy of the PO-CRF models on CoNLL data. A, W, no S/C means: all, word, all but no suffix and cluster features are used, respectively.

included the better the results gets.

For the features, we used word identity, suffixes of up to length 3, Brown cluster and three indicators of (1) capitalization for the first character, (2) containing a hyphen or (3) a digit. Especially, Brown clusters was induced from more than 2 million line documents, making the setting unrealistic for resource-poor language.

With just the word features, the averaged performance is 0.6993 and other indicator features increase the performance to 0.7768. Also note that the suffix and Brown cluster features increase the performance from 0.7768 to 0.8314. As reported, PO-CRF mitigates the adverse effects of the unseen word issues and almost meets the performance in the previous experiment (0.8375) of Täckström et al. (2013) by using these features.

In fourth and final experiment, we used the same features for PO-CRF, with Brown clusters induced on a realistically obtainable sized (3k) corpus for a low resource language. We compare directly to our CCA+SVM model (which does not use Brown clustering features at all). We achieved 0.7983 on PO-CRF with all features and our corresponding model on CCA achieved about 0.8474, shown in Table 5. As reported, our model outperforms the PO-CRF with the realistic settings for resource poor languages.

	PO-CRF 3k Brown	CCA+SVM
BG	0.8318	0.8815
CS	0.7635	0.8232
DA	0.7335	0.8911
DE	0.8296	0.8543
ES	0.8319	0.8713
EU	0.8376	0.8734
HU	0.7817	0.8372
IT	0.8451	0.8474
NL	0.7626	0.8245
PT	0.8768	0.8823
TR	0.6874	0.7354
AVG	0.7983	0.8474

Table 5: Performances on our test data, CoNLL document.

5 Conclusions

We addressed the challenge of POS tagging low-resource languages. Our key idea is to use a massively multilingual corpus. Instead of relying on a single resource-rich language, we leverage the full array of currently available POS taggers. This removes alignment-mismatch noise and identifies a subset of words with highly confident tags. We then use a CCA procedure to induce latent feature representations across domains, incorporating word contexts as well as projected tags. We then train an SVM to predict tags.

Experimentally, we show that this procedure yields accuracy of about 85% for languages with nearly no resources available, beating a state-of-the-art partially observed CRF formulation. In the near future, this technique will enable us to release a suite of POS taggers for hundreds of low-resource languages.

References

Francis R Bach and Michael I Jordan. 2005. A probabilistic interpretation of canonical correlation analysis.

MSo Bartlett. 1936. The square root transformation in analysis of variance. *Supplement to the Journal of the Royal Statistical Society*, pages 68–78.

Koby Crammer and Yoram Singer. 2002. On the learnability and design of output codes for multiclass problems. *Machine Learning*, 47(2-3):201–233.

Dipanjan Das and Slav Petrov. 2011. Unsupervised part-of-speech tagging with bilingual graph-based projections. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 600–609. Association for Computational Linguistics.

John DeNero and Klaus Macherey. 2011. Model-based aligner combination using dual decomposition. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*, pages 420–429. Association for Computational Linguistics.

Paramveer S. Dhillon, Jordan Rodu, Dean P. Foster, and Lyle H. Ungar. 2012. Two Step CCA: A new spectral method for estimating vector models of words. In *Proceedings of the 29th International Conference on Machine learning*, ICML'12.

Greg Durrett, Adam Pauls, and Dan Klein. 2012. Syntactic transfer using a bilingual lexicon. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1–11. Association for Computational Linguistics.

Rong-En Fan, Kai-Wei Chang, Cho-Jui Hsieh, Xiang-Rui Wang, and Chih-Jen Lin. 2008. Liblinear: A library for large linear classification. *The Journal of Machine Learning Research*, 9:1871–1874.

Victoria Fossum and Steven Abney. 2005. Automatically inducing a part-of-speech tagger by projecting from multiple source languages across aligned corpora. In *Natural Language Processing–IJCNLP 2005*, pages 862–873. Springer.

Kuzman Ganchev and Dipanjan Das. 2013. Cross-lingual discriminative learning of sequence models with posterior regularization. In *Proceedings of EMNLP*. Association for Computational Linguistics, October.

Kuzman Ganchev, Joao Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *The Journal of Machine Learning Research*, 11:2001–2049.

Harry R Glahn. 1968. Canonical correlation and its relationship to discriminant analysis and multiple regression. *Journal of the atmospheric sciences*, 25(1):23–31.

Harold Hotelling. 1936. Relations between two sets of variates. *Biometrika*, 28(3-4):321–377.

Sham M Kakade and Dean P Foster. 2007. Multi-view regression via canonical correlation analysis. In *Learning Theory*, pages 82–96. Springer.

Young-Bum Kim and Benjamin Snyder. 2012. Universal grapheme-to-phoneme prediction over latin alphabets. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 332–343. Association for Computational Linguistics.

Young-Bum Kim and Benjamin Snyder. 2013a. Optimal data set selection: An application to grapheme-to-phoneme conversion. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 1196–1205. Association for Computational Linguistics.

Young-Bum Kim and Benjamin Snyder. 2013b. Unsupervised consonant-vowel prediction over hundreds of languages. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 1527–1536. Association for Computational Linguistics.

Young-Bum Kim, João V Graça, and Benjamin Snyder. 2011. Universal morphological analysis using structured nearest neighbor prediction. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 322–332. Association for Computational Linguistics.

Young-Bum Kim, Heemoon Chae, Benjamin Snyder, and Yu-Seop Kim. 2014. Training a korean srl system with rich morphological features. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 637–642. Association for Computational Linguistics.

Young-Bum Kim, Minwoo Jeong, Karl Stratos, and Ruhi Sarikaya. 2015a. Weakly supervised slot tagging with partially labeled sequences from web search click logs. In *Proceedings of the Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 84–92. Association for Computational Linguistics.

Young-Bum Kim, Karl Stratos, Xiaohu Liu, and Ruhi Sarikaya. 2015b. Compact lexicon selection with spectral methods. In *Proceedings of Association for Computational Linguistics (ACL)*, pages 806–811. Association for Computational Linguistics.

Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2015c. Pre-training of hidden-unit crfs. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 192–198. Association for Computational Linguistics.

Young-Bum Kim, Karl Stratos, Ruhi Sarikaya, and Minwoo Jeong. 2015d. New transfer learning techniques for disparate label sets. In *Proceedings of the Association for Computational Linguistics (ACL)*, pages 473–482. Association for Computational Linguistics.

Shen Li, Joao V Graça, and Ben Taskar. 2012. Wiki-ly supervised part-of-speech tagging. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1389–1398. Proceedings of Association for Computational Linguistics.

Percy Liang, Ben Taskar, and Dan Klein. 2006. Alignment by agreement. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 104–111. Association for Computational Linguistics.

Percy Liang, Hal Daumé III, and Dan Klein. 2008. Structure compilation: trading structure for features. In *Proceedings of the 25th international conference on Machine learning*, pages 592–599. ACM.

Thomas Mayer and Michael Cysouw. 2012. Language comparison through sparse multilingual word alignment. In *Proceedings of the EACL 2012 Joint Workshop of LINGVIS & UNCLH*, pages 54–62. Association for Computational Linguistics.

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*, pages 62–72. Association for Computational Linguistics.

Robert C Moore. 2014. Fast high-accuracy part-of-speech tagging by independent classifiers. In *Proceedings of COLING*, pages 1165–1176.

Tahira Naseem, Benjamin Snyder, Jacob Eisenstein, and Regina Barzilay. 2009. Multilingual part-of-speech tagging: Two unsupervised approaches. *Journal of Artificial Intelligence Research*, 36(1):341–385.

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

Karl Stratos, Do-kyum Kim, Michael Collins, and Daniel Hsu. 2014. A spectral algorithm for learning class-based n-gram models of natural language. *Proceedings of the Association for Uncertainty in Artificial Intelligence*.

Karl Stratos, Michael Collins, and Daniel Hsu. 2015. Model-based word embeddings from decompositions of count matrices. In *Proceedings of the Annual Meeting of the Association for Computational Linguistics*, pages 1282–1291. Association for Computational Linguistics, July.

Oscar Täckström, Dipanjan Das, Slav Petrov, Ryan McDonald, and Joakim Nivre. 2013. Token and type constraints for cross-lingual part-of-speech tagging. *Transactions of the Association for Computational Linguistics*, 1:1–12.

Guillaume Wisniewski, Nicolas Pécheux, Souhir Gahbiche-Braham, and François Yvon. 2014. Cross-lingual part-of-speech tagging through ambiguous learning. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1779–1785.

David Yarowsky, Grace Ngai, and Richard Wicentowski. 2001. Inducing multilingual text analysis tools via robust projection across aligned corpora. In *Proceedings of the first international conference on Human language technology research*, pages 1–8. Association for Computational Linguistics.