A Bayesian Mixture Model for Part-of-Speech Induction Using Multiple Features

Christos Christodoulopoulos
School of Informatics
University of Edinburgh
christos.c@ed.ac.uk

Sharon Goldwater
School of Informatics
University of Edinburgh
sgwater@inf.ed.ac.uk

Mark Steedman
School of Informatics
University of Edinburgh
steedman@inf.ed.ac.uk

Abstract

In this paper we present a fully unsupervised syntactic class induction system formulated as a Bayesian multinomial mixture model, where each word type is constrained to belong to a single class. By using a mixture model rather than a sequence model (e.g., HMM), we are able to easily add multiple kinds of features, including those at both the type level (morphology features) and token level (context and alignment features, the latter from parallel corpora). Using only context features, our system yields results comparable to state-of-the art, far better than a similar model without the one-class-per-type constraint. Using the additional features provides added benefit, and our final system outperforms the best published results on most of the 25 corpora tested.

1 Introduction

Research on unsupervised learning for NLP has become widespread recently, with part-of-speech induction, or syntactic class induction, being a particularly popular task.[1] However, despite a recent proliferation of syntactic class induction systems (Biemann, 2006; Goldwater and Griffiths, 2007; Johnson, 2007; Ravi and Knight, 2009; Berg-Kirkpatrick et al., 2010; Lee et al., 2010), careful comparison indicates that very few systems perform better than some much simpler and quicker methods dating back ten or even twenty years (Christodoulopoulos et al., 2010). This fact suggests that we should consider which features of the older systems led to their success, and attempt to combine these features with some of the machine learning methods introduced by the more recent systems. We pursue this strategy here, developing a system based on Bayesian methods where the probabilistic model incorporates several insights from previous work.

Perhaps the most important property of our model is that it is *type-based*, meaning that all tokens of a given word type are assigned to the same cluster. This property is not strictly true of linguistic data, but is a good approximation: as Lee et al. (2010) note, assigning each word type to its most frequent part of speech yields an upper bound accuracy of 93% or more for most languages. Since this is much better than the performance of current unsupervised syntactic class induction systems, constraining the model in this way seems likely to improve performance by reducing the number of parameters in the model and incorporating useful linguistic knowledge. Both of the older systems discussed by Christodoulopoulos et al. (2010), i.e., Clark (2003) and Brown et al. (1992), included this constraint and achieved very good performance relative to token-based systems. More recently, Lee et al. (2010) presented a new type-based model, and also reported very good results.

A second property of our model, which distinguishes it from the type-based Bayesian model of Lee et al. (2010), is that the underlying probabilistic model is a *clustering model*, (specifically, a multinomial mixture model) rather than a sequence model (HMM). In this sense, our model is more closely re-

[1] The task is more commonly referred to as part-of-speech induction, but we prefer the term syntactic class induction since the induced classes may not coincide with part-of-speech tags.

lated to several non-probabilistic systems that cluster context vectors or lower-dimensional representations of them (Redington et al., 1998; Schütze, 1995; Lamar et al., 2010). Sequence models are by far the most common method of supervised part-of-speech tagging, and have also been widely used for unsupervised part-of-speech tagging both with and without a dictionary (Smith and Eisner, 2005; Haghighi and Klein, 2006; Goldwater and Griffiths, 2007; Johnson, 2007; Ravi and Knight, 2009; Lee et al., 2010). However, systems based on context vectors have also performed well in these latter scenarios (Schütze, 1995; Lamar et al., 2010; Toutanova and Johnson, 2007) and present a viable alternative to sequence models.

One advantage of using a clustering model rather than a sequence model is that the features used for clustering need not be restricted to context words. Additional types of features can easily be incorporated into the model and inference procedure using the same general framework as in the basic model that uses only context word features. In particular, we present two extensions to the basic model. The first uses *morphological features*, which serve as cues to syntactic class and seemed to partly explain the success of two best-performing systems analysed by Christodoulopoulos et al. (2010). The second extension to our model uses *alignment features* gathered from parallel corpora. Previous work suggests that using parallel text can improve performance on various unsupervised NLP tasks (Naseem et al., 2009; Snyder and Barzilay, 2008).

We evaluate our model on 25 corpora in 20 languages that vary substantially in both syntax and morphology. As in previous work (Lee et al., 2010), we find that the one-class-per-type restriction boosts performance considerably over a comparable token-based model and yields results that are comparable to state-of-the-art even without the use of morphology or alignment features. Including morphology features yields the best published results on 14 or 15 of our 25 corpora (depending on the measure) and alignment features can improve results further.

2 Models

Our model is a multinomial mixture model with Bayesian priors over the mixing weights θ and multinomial class output parameters ϕ. The model is defined so that all observations associated with a single word type are generated from the same mixing component (syntactic class). In the basic model, these observations are token-level features; the morphology model adds type-level features as well. We begin by describing the simplest version of our model, where each word token is associated with a single feature, for example its left context word (the word that occurs to its left in the corpus). We then show how to generalise the model to multiple token-level features and to type-level features.

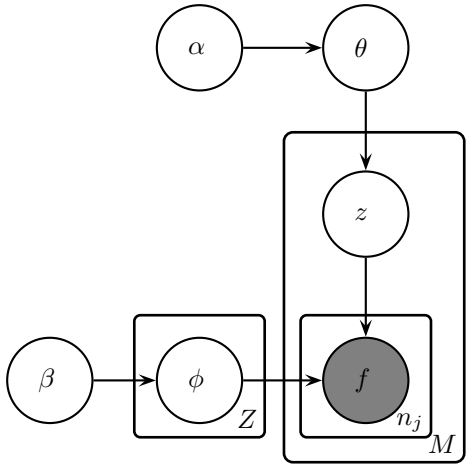

Figure 1: Plate diagram of the basic model with a single feature per token (the observed variable f). M, Z, and n_j are the number of word types, syntactic classes z, and features (= tokens) per word type, respectively.

2.1 Basic model

In the basic model, each word token is represented by a single feature such as its left context word. These features are the observed data; the model explains the data by assuming that it has been generated from some set of latent syntactic classes. The ith class is associated with a multinomial parameter vector ϕ_i that defines the distribution over features generated from that class, and with a mixing weight θ_i that defines the prior probability of that class. θ and ϕ_i are drawn from symmetric Dirichlet distributions with parameters α and β respectively.

The generative story goes as follows: First, generate the prior class probabilities θ. Next, for each

word type $j = 1 \ldots M$, choose a class assignment z_j from the distribution θ. For each class $i = 1 \ldots Z$, choose an output distribution over features ϕ_i. Finally, for each token $k = 1 \ldots n_j$ of word type j, generate a feature f_{jk} from ϕ_{z_j}, the distribution associated with the class that word type j is assigned to. The model is illustrated graphically in Figure 1 and is defined formally as follows:

$$\begin{aligned} \theta \,|\, \alpha &\sim \text{Dirichlet}(\alpha) \\ z_j \,|\, \theta &\sim \text{Multinomial}(\theta) \\ \phi_i \,|\, \beta &\sim \text{Dirichlet}(\beta) \\ f_{jk} \,|\, \phi_{z_j} &\sim \text{Multinomial}(\phi_{z_j}) \end{aligned}$$

In addition to the variables defined above, we will use F to refer to the number of different possible values a feature can take on (so that ϕ is a $Z \times F$ matrix). Thus, one way to think of the model is as a vector-based clustering system, where word type j is associated with a $1 \times F$ vector of feature counts representing the features of all n_j tokens of j, and these vectors are clustered into similar classes. The difference from other vector-based syntactic class induction systems is in the method of clustering. Here, we define a Gibbs sampler that samples from the posterior distribution of the clusters given the observed features; other systems have used various standard distance-based vector clustering methods. Some systems also include dimensionality reduction (Schütze, 1995; Lamar et al., 2010) to reduce the size of the context vectors; we simply use the F most common words as context features.

2.2 Inference

At inference time we want to sample a syntactic class assignment $\mathbf{z}$ from the posterior of the model. We use a collapsed Gibbs sampler, integrating out the parameters θ and ϕ and sampling from the following distribution:

$$P(\mathbf{z}|\mathbf{f}, \alpha, \beta) \propto P(\mathbf{z}|\alpha)P(\mathbf{f}|\mathbf{z}, \beta). \quad (1)$$

Rather than sampling the joint class assignment $P(\mathbf{z}|\mathbf{f}, \alpha, \beta)$ directly, the sampler iterates over each word type j, resampling its class assignment z_j given the current assignments $\mathbf{z}_{-j}$ of all other word types. The posterior over z_j can be computed as

$$\begin{aligned} &P(z_j \,|\, \mathbf{z}_{-j}, \mathbf{f}, \alpha, \beta) \\ &\quad \propto \quad P(z_j \,|\, \mathbf{z}_{-j}, \alpha, \beta) P(\mathbf{f}_j \,|\, \mathbf{f}_{-j}, \mathbf{z}, \alpha, \beta) \end{aligned} \quad (2)$$

where $\mathbf{f}_j$ are the features associated with word type j (one feature for each token of j). The first (prior) factor is easy to compute due to the conjugacy between the Dirichlet and multinomial distributions, and is equal to

$$P(z_j = z \,|\, \mathbf{z}_{-j}, \alpha) = \frac{n_z + \alpha}{n_\cdot + Z\alpha} \quad (3)$$

where n_z is the number of types in class z and $n_\cdot$ is the total number of word types in all classes. All counts in this and the following equations are computed with respect to $\mathbf{z}_{-j}$ (e.g., $n_\cdot = M - 1$).

Computing the second (likelihood) factor is slightly more complex due to the dependencies between the different variables in $\mathbf{f}_j$ that are induced by integrating out the ϕ parameters. Consider first a simple case where word type j occurs exactly twice in the corpus, so $\mathbf{f}_j$ contains two features. The probability of the first feature f_{j1} is equal to

$$P(f_{j1} = f \,|\, z_j = z, \mathbf{z}_{-j}, \mathbf{f}_{-j}, \beta) = \frac{n_{f,z} + \beta}{n_{\cdot,z} + F\beta} \quad (4)$$

where $n_{f,z}$ is the number of times feature f has been seen in class z, $n_{\cdot,z}$ is the total number of feature tokens in the class, and F is the number of different possible features.

The probability of the second feature f_{j2} can be calculated similarly, except that it is conditioned on f_{j1} in addition to the other variables, so the counts for previously observed features must include the counts due to f_{j1} as well as those due to $\mathbf{f}_{-j}$. Thus, the probability is

$$\begin{aligned} &P(f_{j2} = f \,|\, f_{j1}, z_j = z, \mathbf{z}_{-j}, \mathbf{f}_{-j}, \beta) \\ &\qquad = \frac{n_{f,z} + \delta(f_{j1}, f_{j2}) + \beta}{n_{\cdot,z} + 1 + F\beta} \end{aligned} \quad (5)$$

where δ is the Kronecker delta function, equal to 1 if its arguments are equal and 0 otherwise.

Extending this example to the general case, the probability of a sequence of features $\mathbf{f}_j$ is computed using the chain rule, where the counts used in each factor are incremented as necessary for each additional conditioning feature, yielding the following expression:

$$\begin{aligned} &P(\mathbf{f}_j \,|\, \mathbf{f}_{-j}, z_j = z, \mathbf{z}_{-j}, \beta) \\ &\qquad = \frac{\prod_{k=1}^{F} \prod_{i=0}^{n_{jk}-1} (n_{jk,z} + i + \beta)}{\prod_{i=0}^{n_j-1} (n_{\cdot,z} + i + F\beta)} \end{aligned} \quad (6)$$

where n_{jk} is the number of instances of feature k in word type j.[2]

2.3 Extended models

We can extend the model above in two different ways: by adding more features at the word token level, or by adding features at the type level. To add more token-level features, we simply assume that each word token generates multiple features, one feature from each of several different kinds.[3] For example, the left context word might be one kind of feature and the right context word another. We assume conditional independence between the generated features given the syntactic class, so each kind of feature t has its own output parameters $\phi^{(t)}$. A plate diagram of the model with T kinds of features is shown in Figure 2 (a type-level feature is also included in this diagram, as described below).

Due to the independence assumption between the different kinds of features, the basic Gibbs sampler is easy to extend to this case by simpling multiplying in extra factors for the additional kinds of features, with the prior (Equation 3) unchanged. The likelihood becomes:

$$P(\mathbf{f}_j^{(1)}, \ldots, \mathbf{f}_j^{(T)} \mid \mathbf{f}_{-j}^{(1\ldots T)}, z_j = z, \mathbf{z}_{-j}, \beta) = \prod_{t=1}^{T} P(\mathbf{f}_j^{(t)} \mid \mathbf{f}_{-j}^{(t)}, z_j = z, \mathbf{z}_{-j}, \beta) \quad (7)$$

where each factor in the product is computed using Equation 6.

In addition to monolingual context features, we also explore the use of alignment features for those languages where we have parallel corpora. These features are extracted for language ℓ by word-aligning ℓ to another language ℓ' (details of the alignment procedure are described in Section 3.1). The features used for each token e in ℓ are the left and right context words of the word token that is aligned to e (if there is one). As with the monolingual context features, we use only the F most frequent words in ℓ' as possible features.

[2]One could approximate this likelihood term by assuming independence between all n_j feature tokens of word type j. This is the approach taken by Lee et al. (2010).

[3]We use the word *kind* here to avoid confusion with *type*, which we reserve for the type-token distinction, which can apply to features as well as words.

Note that this model with multiple context features is deficient: it can generate data that are inconsistent with any actual corpus, because there is no mechanism to constrain the left context word of token e_i to be the same as the right context word of token e_{i-1} (and similarly with alignment features). However, deficient models have proven useful in other unsupervised NLP tasks (Klein and Manning, 2002; Toutanova and Johnson, 2007). In particular, Toutanova and Johnson (2007) demonstrate good performance on unsupervised part-of-speech tagging (using a dictionary) with a Bayesian model similar to our own. If we remove the part of their model that relies on the dictionary (the morphological ambiguity classes), their model is equivalent to our own, without the restriction of one class per type. We use this token-based version of our model as a baseline in our experiments.

The final extension to our model introduces type-level features, specifically morphology features. The model is illustrated in Figure 2. We assume conditional independence between the morphology features and other features, so again we can simply multiply another factor into the likelihood during inference. There is only one morphological feature per type, so this factor has the form of Equation 4. Since frequent words will have many token-level features contributing to the likelihood and only one morphology feature, the morphology features will have a greater effect for infrequent words (as appropriate, since there is less evidence from context and alignments). As with the other kinds of features, we use only a limited number F_m of morphology features, as described below.

3 Experiments

3.1 Experimental setup

We evaluate our models using an increasing level of complexity, starting with a model that uses only monolingual context features. We use the $F = 100$ most frequent words as features, and consider two versions of this model: one with two kinds of features (one left and one right context word) and one with four (two context words on each side).

For the model with morphology features we ran the unsupervised morphological segmentation system Morfessor (Creutz and Lagus, 2005) to get a

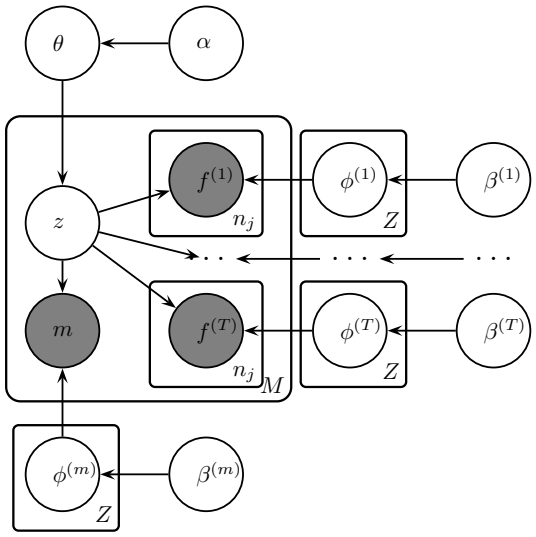

Figure 2: Plate diagram of the extended model with T kinds of token-level features ($f^{(t)}$ variables) and a single kind of type-level feature (morphology, m).

segmentation for each word type in the corpus. We then extracted the suffix of each word type[4] and used it as a feature type. This process yielded on average $F_m = 110$ morphological feature types[5]. Each word type generates at most one of these possible features. If there are overlapping possibilities (e.g. -ingly and -y) we take the longest possible match.

We also explore the idea of extending the morphology feature space beyond suffixes, by including features like capitalisation and punctuation. Specifically we use the features described in Haghighi and Klein (2006), namely *initial-capital*, *contains-hyphen*, *contains-digit* and we add an extra feature *contains-punctuation*.

For the model with alignment features, we follow (Naseem et al., 2009) in using only bidirectional alignments: using Giza++ (Och and Ney, 2003), we get the word alignments in both directions between all possible language pairs in our parallel corpora (i.e., alternating the source and target languages within each pair). We then use only those alignments that are found in both directions. As discussed above, we use two kinds of alignment features: the left and right context words of the aligned token in the other language. The feature space is set to the $F = 100$ most frequent words in that language.

Instead of fixing the hyperparameters α and β, we used the Metropolis-Hastings sampler presented by Goldwater and Griffiths (2007) to get updated values based on the likelihood of the data with respect to those hyperparameters[6]. In order to improve convergence of the sampler, we used simulated annealing with a sigmoid-shaped cooling schedule from an initial temperature of 2 down to 1. Preliminary experiments indicated that we could achieve better results by cooling even further (approximating the MAP solution rather than a sample from the posterior), so for all experiments reported here, we ran the sampler for a total of 2000 iterations, with the last 400 of these decreasing the temperature from 1 to 0.66.

Finally, we investigated two different initialisation techniques: First, we use random class assignments to word types (referred to as method 1) and second, we assign each of the Z most frequent word types to a separate class and then randomly distribute the rest of the word types to the classes (method 2).

3.2 Datasets

Although unsupervised systems should in principle be language- and corpus-independent, most part-of-speech induction systems (especially in the early literature) have been developed on English. Whether because English is simply an easier language, or because of bias introduced during development, these systems' performance is considerably worse in other languages (Christodoulopoulos et al., 2010)

Since we aim to use our system mostly on non-English corpora, and ones that are significantly smaller than the large English treebank corpora, we developed our models using one of the languages of the MULTEXT-East corpus (Erjavec, 2004), namely Bulgarian. The other languages in the corpus were used during development as a source of word alignments, but otherwise were only used for testing final versions of our models. Since none of the authors speak any of the languages in the MULTEXT col-

[4]Since Morfessor yields multiple affixes for each word we concatenated all the suffixes into a single suffix.

[5]There was large variance in the number of feature types for each language ranging from 11 in Chinese to more than 350 in German and Czech.

[6]For simplicity, we tied the β parameters for the two or four kinds of context features to the same value, and similarly the β parameters for the two kinds of alignment features.

lection, we also used the Penn Treebank WSJ corpus (Marcus et al., 1993) for development. Following Christodoulopoulos et al. (2010) we created a smaller version of the WSJ corpus (referred to as wsj-s) to approximate the size of the corpora in MULTEXT-East. For comparison to other systems, we also used the full WSJ at test time.

For further testing, we used the remaining MULTEXT languages, as well as the languages of the CONNL-X (Buchholz and Marsi, 2006) shared task. This dataset contains 13 languages, 4 of which are freely available (Danish, Dutch, Portuguese and Swedish) and 9 that are used with permission from the creators of the corpora (Arabic[7], Bulgarian[8], Czech[9], German[10], Chinese[11], Japanese[12], Slovene[13], Spanish[14], Turkish[15]). Following Lee et al. (2010) we used only the training sections for each language.

Finally, to widen the scope of our system, we generated two more corpora in French[16] and Ancient Greek[17], extracting the gold standard parts of speech from the respective dependency treebanks.

3.3 Baselines

We chose three baselines for comparison. The first is the basic k-means clustering algorithm, which we applied to the same feature vectors we extracted for our system (context + extended morphology), using a Euclidean distance metric. This provides a very simple vector-based clustering baseline. The second baseline is a more recent vector-based syntactic class induction method, the SVD approach of (Lamar et al., 2010), which extends Schütze (1995)'s original method and, like ours, enforces a one-class-per-tag restriction. As a third baseline we use the system of Clark (2003) since it is a type-level system that models morphology and has produced very good results on multilingual corpora.

[7]Part of the Prague Arabic Treebank (Hajič et al., 2003; Smrž and Pajas, 2004)

[8]Part of the BulTreeBank (Simov et al., 2004).

[9]Part of the Prague Dep. Treebank (Böhmová et al., 2001)

[10]Part of the TIGER Treebank (Brants et al., 2002)

[11]Part of the Sinica Treebank (Keh-Jiann et al., 2003)

[12]Part of the Tübingen Treebank of Spoken Japanese (formerly VERMOBIL Treebank - Kawata and Bartels (2000)).

[13]Part of the Slovene Dep. Treebank (Džeroski et al., 2006)

[14]Part of the Cast3LB Treebank (Civit et al., 2006)

[15]Part of the METU-Sabanci Treebank (Oflazer et al., 2003).

[16]French Treebank (Abeillé et al., 2000)

[17]Greek Dependency Treebank (Bamman et al., 2009)

4 Results and Analysis

4.1 Development results

Tables 1 and 2 present the results from development runs, which were used to decide which features to incorporate in the final system. We used V-Measure (Rosenberg and Hirschberg, 2007) as our primary evaluation score, but also present many-to-one matching accuracy (M-1) scores for better comparison with previously published results. We chose V-Measure (VM) as our evaluation score because it is less sensitive to the number of classes induced by the model (Christodoulopoulos et al., 2010), allowing us to develop our models without using the number of classes as a parameter. We fixed the number of classes in all systems to 45 during development; note however that the gold standard tag set for Bulgarian contains only 12 tags, so the results in Table 1 (especially the M-1 scores) are not comparable to previous results. For results using the number of gold-standard tags refer to Table 4.

The first conclusion that can be drawn from these results is the large difference between the token- and type-based versions of our system, which confirms that the one-class-per-type restriction is helpful for unsupervised syntactic class induction. We also see that for both languages, the performance of the model using 4 context words (± 2 on each side) is worse than the 2 context words model. We therefore used only two context words for all of our additional test languages (below).

We can clearly see that morphological features are helpful in both languages; however the extended features of Haghighi and Klein (2006) seem to help only on the English data. This could be due to the fact that Bulgarian has a much richer morphology and thus the extra features contribute little to the overall performance of the model.

The contribution of the alignment features on the Bulgarian corpus (aligned with English) is less significant than that of morphology but when combined, the two sets of features yield the best performance. This provides evidence in favor of using multiple features.

Finally, initialisation method 2 does not yield

system	±1 words VM/M-1	±2 words VM/M-1
base	58.1 / 70.8	55.4 / 67.6
base(tokens)	48.3 / 62.5	37.0 / 54.4
base(init)	57.6 / 70.1	56.1 / 68.6
+morph	58.3 / 74.9	57.4 / 71.9
+morph(ext)	57.8 / 73.7	57.8 / 70.1
(init)+morph	57.8 / 74.3	57.3 / 69.5
(init)+morph(ext)	58.1 / 74.3	57.2 / 71.3
+aligns(EN)	58.1 / 72.6	56.7 / 71.1
+aligns(EN)+morph	**59.0** / **75.4**	57.5 / 69.7

Table 1: V-measure (VM) and many-to-one (M-1) results on the MULTEXT-Bulgarian corpus for various models using either ±1 or ±2 context words as features. base: context features only; (tokens): token-based model; (init): Initialisation method 2—other results use method 1; (ext): Extended morphological features.

system	±1 words VM/M-1	±2 words VM/M-1
base	63.3 / 64.3	62.4 / 63.3
base(tokens)	48.6 / 57.8	49.3 / 38.3
base(init)	62.7 / 62.9	62.2 / 62.4
+morph	66.4 / 66.7	65.1 / 67.2
+morph(ext)	**67.7** / **72.0**	65.6 / 67.0
(init)+morph	64.8 / 66.9	64.2 / 66.0
(init)+morph(ext)	67.4 / 71.3	65.7 / 67.1

Table 2: V-measure and many-to-one results on the wsj-s corpus for various models, as described in Table 1.

.

consistent improvements over the standard random initialisation—if anything, it seems to perform worse. We therefore use only method 1 in the remaining experiments.

4.2 Overall results

Table 3 presents the results on our parallel corpora. We tested all possible combinations of two languages to align, and present both the average score over all alignments, and the score under the best choice of aligned language.[18] Also shown are the results of adding morphology features to the basic model (context features only) and to the best alignment model for each language. In accord with our development results, adding morphology to the basic model is generally useful. The alignment results are mixed: on the one hand, choosing the best possible language to align yields improvements, which can be improved further by adding morphological features, resulting in the best scores of all models for most languages. On the other hand, without knowing which language to choose, alignment features do not help on average. We note, however, that three out of the seven languages have English as their best-aligned pair (perhaps due to its better overall scores), which suggests that in the absence of other knowledge, aligning with English may be a good choice.

The low average performance of the alignment features is disappointing, but there are many possible variations on our method for extracting these features that we have not yet tested. For example, we used only bidirectional alignments in an effort to improve alignment precision, but these alignments typically cover less than 40% of tokens. It is possible that a higher-recall set of alignments could be more useful.

We turn now to our results on all 25 corpora, shown in Table 4 along with corpus statistics, baseline results, and the best published results for each language (when available). Our system, including morphology features in all cases, is listed as BMMM (Bayesian Multinomial Mixture Model). We do not include alignment features for the MULTEXT languages since these features only yielded improvements for the oracle case where we know which aligned language to choose. Nevertheless, our MULTEXT scores mostly outperform all other systems. Overall, we acheive the highest published results on 14 (VM) or 15 (M-1) of the 25 corpora.

One surprising discovery is the high performance of the k-means clustering system. Despite its simplicity, it is competitive with the other systems and in a few cases even achieves the best published results.

5 Conclusion

We have presented a Bayesian model for syntactic class induction that has two important properties. First, it is type-based, assigning the same class to every token of a word type. We have shown by

[18] The choice of language was based on the same test data, so the 'best-language' results should be viewed as oracle scores.

Lang.	BASE		ALIGNMENTS		
	base	**+morph**	**Avg.**	**Best**	**+morph**
	VM/M-1	VM/M-1	VM/M-1	VM/M1	VM/M1
Bulgarian	54.4 / 61.5	54.5 / 64.3	53.1 / 60.5	55.2 / 64.5(EN)	**55.7** / **66.0**
Czech	54.2 / 58.9	53.9 / 64.2	52.6 / 58.4	53.8 / 59.7(EN)	**55.4** / **66.4**
English	62.9 / 72.4	63.3 / 73.3	62.5 / 72.0	63.2 / 71.9(HU)	**63.5** / **73.7**
Estonian	52.8 / 63.5	53.3 / **67.4**	52.8 / 63.9	53.5 / 65.0(EN)	**54.3** / 66.9
Hungarian	53.3 / 60.4	54.8 / **68.2**	53.3 / 60.8	53.9 / 61.1(RO)	**55.9** / 67.1
Romanian	53.9 / 62.4	52.3 / 61.1	56.2 / 63.7	**57.5** / **64.6**(ES)	54.5 / 63.4
Slovene	**57.2** / 65.9	56.7 / 67.9	54.7 / 64.1	55.9 / 64.4(HU)	56.7 / **67.9**
Serbian	**49.1** / 56.6	49.0 / **62.0**	47.3 / 55.6	48.9 / 59.4(CZ)	48.3 / 60.8

Table 3: V-measure (VM) and many-to-one (M-1) results on the languages in the MULTEXT-East corpus using the gold standard number of classes shown in Table 4. BASE results use $\pm$1-word context features alone or with morphology. ALIGNMENTS adds alignment features, reporting the average score across all possible choices of paired language and the scores under the best performing paired language (in parens), alone or with morphology features.

	Language	Types	Tags	k-means	SVD2	clark	Best Pub.	BMMM
WSJ	wsj	49,190	45	59.5 / 61.6	58.2 / 64.0	65.6 / 71.2	68.8 / 76.1*	66.1 / 72.8
	wsj-s	16,850	45	56.7 / 60.1	54.3 / 60.7	63.8 / 68.8	62.3 / 70.7*	**67.7** / **72.0**
MULTEXT-East	Bulgarian	16,352	12	50.3 / 59.3	41.7 / 51.0	**55.6** / **66.5**	-	54.5 / 64.4
	Czech	19,115	12	48.6 / 56.7	35.5 / 50.9	52.6 / 64.1	-	**53.9** / **64.2**
	English	9,773	12	56.5 / 65.4	52.3 / 65.5	60.5 / 70.6	-	**63.3** / **73.3**
	Estonian	17,845	11	45.3 / 55.6	38.7 / 55.3	44.4 / 58.4	-	**53.3** / **64.4**
	Hungarian	20,321	12	46.7 / 53.9	39.8 / 49.5	48.9 / 61.4	-	**54.8** / **68.2**
	Romanian	15,189	14	45.2 / 55.1	42.1 / 52.6	40.9 / 49.9	-	**52.3** / **61.1**
	Slovene	17,871	12	46.9 / 56.2	39.5 / 54.2	54.9 / 69.4	-	**56.7** / **67.9**
	Serbian	18,095	12	41.4 / 47.0	39.1 / 54.6	**51.0** / **64.1**	-	49.0 / 62.0
CoNLL06 Shared Task	Arabic	12,915	20	**43.3** / 60.7	27.6 / 49.0	40.6 / 59.8	-	42.4 / **61.5**
	Bulgarian	32,439	54	53.6 / 65.6	49.0 / 65.3	**59.6** / **70.4**	-	58.8 / 68.9
	Chinese	40,562	15	32.6 / 61.1	24.5 / 54.6	31.8 / 56.7	-	**42.6** / **69.4**
	Czech	130,208	12	-	-	47.1 / 65.5	-	**48.4** / **65.7**
	Danish	18,356	25	51.7 / 61.6	40.8 / 57.6	52.7 / 65.3	- / 66.7†	**59.0** / **71.1**
	Dutch	28,393	13	45.3 / 60.5	36.7 / 52.4	52.2 / 67.9	- / 67.3‡	**54.7** / **71.1**
	German	72,326	54	58.7 / 67.5	54.1 / 64.2	**63.0** / 73.9	- / 68.4‡	61.9 / **74.4**
	Japanese	3,231	80	76.1 / 76.2	74.4 / 75.5	**78.6** / 77.4	-	77.4 / **78.5**
	Portuguese	28,931	22	51.6 / 64.4	45.9 / 63.1	57.4 / 69.2	- / 75.3†	**63.9** / **76.8**
	Slovene	7,128	29	52.6 / **64.2**	44.0 / 60.3	**53.9** / 63.5	-	49.4 / 56.2
	Spanish	16,458	47	59.5 / 69.2	54.8 / 68.2	61.6 / 71.9	- / **73.2**†	**63.2** / 71.7
	Swedish	20,057	41	53.2 / 62.2	47.4 / 59.1	**58.9** / **68.7**	- / 60.6‡	58.0 / 68.2
	Turkish	17,563	30	**40.8** / **62.8**	27.4 / 52.4	36.8 / 58.1	-	40.2 / 58.7
	French	49,964	23	48.2 / 68.6	46.3 / 68.5	**57.3** / **77.8**	-	55.0 / 76.6
	A.Greek	15,194	15	38.6 / 44.8	24.2 / 38.5	33.3 / **45.4**	-	**40.5** / 45.1

Table 4: Final results on 25 corpora in 20 languages, with the number of induced classes equal to the number of gold standard tags in all cases. **k-means** and **SVD2** models could not produce a clustering in the Czech CoNLL corpus due its size. Best published results are from *Christodoulopoulos et al. (2010), †Berg-Kirkpatrick et al. (2010) and ‡Lee et al. (2010). The latter two papers do not report VM scores. No best published results are shown for the MULTEXT languages; Christodoulopoulos et al. (2010) report results based on 45 tags suggesting that **clark** performs best on these corpora.

comparison with a token-based version of the model that this restriction is very helpful. Second, it is a clustering model rather than a sequence model. This property makes it easy to incorporate multiple kinds of features into the model at either the token or the type level. Here, we experimented with token-level context features and alignment features and type-level morphology features, showing that morphology features are helpful in nearly all cases, and alignment features can be helpful if the aligned language is properly chosen. Our results even without these extra features are competitive with state-of-the-art; with the additional features we achieve the best published results in the majority of the 25 corpora tested.

Since it is so easy to add extra features to our model, one direction for future work is to explore other possible features. For example, it could be useful to add dependency features from an unsupervised dependency parser. We are also interested in improving our morphology features, either by considering other ways to extract features during preprocessing (for example, including prefixes or not concatenating together all suffixes), or by developing a joint model for inducing both morphology and syntactic classes simultaneously. Finally, our model could be extended by replacing the standard mixture model with an infinite mixture model (Rasmussen, 2000) in order to induce the number of syntactic classes automatically.

Acknowledgments

The authors would like to thank Emily Thomforde, Ioannis Konstas, Tom Kwiatkowski and the anonymous reviewers for their comments and suggestions. We would also like to thank Kiril Simov, Toni Marti, Tomaz Erjavec, Jess Lin and Kathrin Beck for providing us with CoNLL data. This work was supported by an EPSRC graduate Fellowship, and by ERC Advanced Fellowship 249520 GRAMPLUS.

References

Anne Abeillé, Lionel Clément, and Alexandra Kinyon. 2000. Building a treebank for French. In *In Proceedings of the LREC 2000*.

David Bamman, Francesco Mambrini, and Gregory Crane. 2009. An ownership model of annotation: The Ancient Greek dependency treebank. In *TLT 2009-Eighth International Workshop on Treebanks and Linguistic Theories*.

Taylor Berg-Kirkpatrick, Alexandre B. Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Proceedings of NAACL 2010*, pages 582–590, Los Angeles, California, June.

Chris Biemann. 2006. Unsupervised part-of-speech tagging employing efficient graph clustering. In *Proceedings of COLING ACL 2006*, pages 7–12, Morristown, NJ, USA.

Alena Böhmová, Jan Hajič, Eva Hajičová, and Barbora Hladká. 2001. The Prague dependency treebank: Three-level annotation scenario. In Anne Abeillé, editor, *Treebanks: Building and Using Syntactically Annotated Corpora*, pages 103 – 126. Kluwer Academic Publishers.

Sabine Brants, Stefanie Dipper, Silvia Hansen, Wolfgang Lezius, and George Smith. 2002. The TIGER treebank. In *Proceedings of the Workshop on Treebanks and Linguistic Theories*, Sozopol.

Peter F. Brown, Vincent J. Della Pietra, Peter V. Desouza, Jennifer C. Lai, and Robert L. Mercer. 1992. Class-based n-gram models of natural language. *Computational Linguistics*, 18(4):467–479.

Sabine Buchholz and Erwin Marsi. 2006. CoNLL-X shared task on multilingual dependency parsing. In *Proceedings of the Tenth Conference on Computational Natural Language Learning*, CoNLL-X '06, pages 149–164, Stroudsburg, PA, USA. Association for Computational Linguistics.

Christos Christodoulopoulos, Sharon Goldwater, and Mark Steedman. 2010. Two decades of unsupervised POS induction: How far have we come? In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, pages 575–584, Cambridge, MA, October. Association for Computational Linguistics.

Montserrat Civit, Ma. Martí, and Núria Bufí. 2006. Cat3lb and cast3lb: From constituents to dependencies. In Tapio Salakoski, Filip Ginter, Sampo Pyysalo, and Tapio Pahikkala, editors, *Advances in Natural Language Processing*, volume 4139 of *Lecture Notes in Computer Science*, pages 141–152. Springer Berlin / Heidelberg.

Alexander Clark. 2003. Combining distributional and morphological information for part of speech induction. In *Proceedings of EACL 2003*, pages 59–66, Morristown, NJ, USA.

Mathias Creutz and Krista Lagus. 2005. Inducing the morphological lexicon of a natural language from unannotated text. In *In Proceedings of the International and Interdisciplinary Conference on*

Adaptive Knowledge Representation and Reasoning (AKRR'05), volume 5, pages 106–113.

Sašo Džeroski, Tomaž Erjavec, Nina Ledinek, Petr Pajas, Zdenek Žabokrtsky, and Andreja Žele. 2006. Towards a Slovene dependency treebank. In *Proceedings Int. Conf. on Language Resources and Evaluation (LREC)*.

Tomaž Erjavec. 2004. MULTEXT-East version 3: Multilingual morphosyntactic specifications, lexicons and corpora. In *Fourth International Conference on Language Resources and Evaluation, (LREC'04)*, pages 1535 – 1538, Paris. ELRA.

Sharon Goldwater and Tom Griffiths. 2007. A fully bayesian approach to unsupervised part-of-speech tagging. In *Proceedings of ACL 2007*, pages 744–751, Prague, Czech Republic, June.

Aria Haghighi and Dan Klein. 2006. Prototype-driven learning for sequence models. In *Proceedings of NAACL 2006*, pages 320–327, Morristown, NJ, USA.

Jan Hajič, Jarmila Panevová, Zdeňka Urešová, Alevtina Bémová, and Petr Pajas. 2003. PDTVALLEX: creating a large-coverage valency lexicon for treebank annotation. In *Proceedings of The Second Workshop on Treebanks and Linguistic Theories*, pages 57–68. Vaxjo University Press.

Mark Johnson. 2007. Why doesn't EM find good HMM POS-taggers? In *Proceedings of EMNLP-CoNLL 2007*, pages 296–305, Prague, Czech Republic, June.

Yasushira Kawata and Julia Bartels. 2000. Stylebook for the Japanese treebank in VERMOBIL. Technical report, Universität Tüubingen.

Chen Keh-Jiann, Chu-Ren Huang, Feng-Yi Chen, Chi-Ching Luo, Ming-Chung Chang, Chao-Jan Chen, and Zhao-Ming Gao. 2003. Sinica treebank: Design criteria, representational issues and implementation. In Anne Abeillé, editor, *Treebanks: Building and Using Syntactically Annotated Corpora*, pages 231–248. Kluwer Academic Publishers.

Dan Klein and Christopher D. Manning. 2002. A generative constituent-context model for improved grammar induction. In *Proceedings of ACL 40*, pages 128–135.

Michael Lamar, Yariv Maron, Mark Johnson, and Elie Bienenstock. 2010. SVD and clustering for unsupervised POS tagging. In *Proceedings of the ACL 2010 Conference Short Papers*, pages 215–219, Stroudsburg, PA, USA. Association for Computational Linguistics.

Yoong Keok Lee, Aria Haghighi, and Regina Barzilay. 2010. Simple type-level unsupervised POS tagging. In *Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing*, EMNLP '10, pages 853–861, Stroudsburg, PA, USA. Association for Computational Linguistics.

Mitchell Marcus, Beatrice Santorini, and Mary Ann Marcinkiewicz. 1993. Building a large annotated corpus of English: the Penn Treebank. *Computational Linguistics*, 19(2):331–330.

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

Franz J. Och and Hermann Ney. 2003. A systematic comparison of various statistical alignment models. *Comput. Linguist.*, 29(1):19–51.

Kemal Oflazer, Bilge Say, Dilek Z. Hakkani-Tür, and Gökhan Tür, 2003. *Building A Turkish Treebank*, chapter 1, pages 1–17. Kluwer Academic Publishers.

Carl Rasmussen. 2000. The infinite Gaussian mixture model. In *Advances in Neural Information Processing Systems 12*.

Sujith Ravi and Kevin Knight. 2009. Minimized models for unsupervised Part-of-Speech tagging. In *Proceedings of ACL-IJCNLP 2009*, pages 504–512, Suntec, Singapore, August.

Martin Redington, Nick Chater, and Steven Finch. 1998. Distributional information: a powerful cue for acquiring syntactic categories. *Cognitive Science*, 22:425 – 469.

Andrew Rosenberg and Julia Hirschberg. 2007. V-measure: A conditional entropy-based external cluster evaluation measure. In *Proceedings of EMNLP-CoNLL 2007*, pages 410–420.

Hinrich Schütze. 1995. Distributional part-of-speech tagging. In *Proceedings of EACL 7*, pages 141–148, San Francisco, CA, USA.

Kiril Simov, Petya Osenova, Alexander Simov, and Milen Kouylekov. 2004. Design and implementation of the Bulgarian HPSG-based treebank. *Research on Language & Computation*, 2(4):495–522.

Noah A. Smith and Jason Eisner. 2005. Contrastive estimation: training log-linear models on unlabeled data. In *Proceedings of ACL 2005*, pages 354–362, Morristown, NJ, USA.

Otakar Smrž and Petr Pajas. 2004. Morphotrees of Arabic and their annotation in the TrEd environment. In *Proceedings of the NEMLAR International Conference on Arabic Language Resources and Tools*, pages 38–41.

Ben Snyder and Regina Barzilay. 2008. Unsupervised multilingual learning for morphological segmentation. In *Proceedings of ACL*.

Kristina Toutanova and Mark Johnson. 2007. A Bayesian LDA-based model for semi-supervised part-of-speech tagging. In *Proceedings of NIPS 2007*.