A Hierarchical Pitman-Yor Process HMM
for Unsupervised Part of Speech Induction

Phil Blunsom
Department of Computer Science
University of Oxford
Phil.Blunsom@cs.ox.ac.uk

Trevor Cohn
Department of Computer Science
University of Sheffield
T.Cohn@dcs.shef.ac.uk

Abstract

In this work we address the problem of unsupervised part-of-speech induction by bringing together several strands of research into a single model. We develop a novel hidden Markov model incorporating sophisticated smoothing using a hierarchical Pitman-Yor processes prior, providing an elegant and principled means of incorporating lexical characteristics. Central to our approach is a new type-based sampling algorithm for hierarchical Pitman-Yor models in which we track fractional table counts. In an empirical evaluation we show that our model consistently out-performs the current state-of-the-art across 10 languages.

1 Introduction

Unsupervised part-of-speech (PoS) induction has long been a central challenge in computational linguistics, with applications in human language learning and for developing portable language processing systems. Despite considerable research effort, progress in fully unsupervised PoS induction has been slow and modern systems barely improve over the early Brown et al. (1992) approach (Christodoulopoulos et al., 2010). One popular means of improving tagging performance is to include supervision in the form of a tag dictionary or similar, however this limits portability and also comprimises any cognitive conclusions. In this paper we present a novel approach to fully unsupervised PoS induction which uniformly outperforms the existing state-of-the-art across all our corpora in 10 different languages. Moreover, the performance of our unsupervised model approaches that of many existing semi-supervised systems, despite our method not receiving any human input.

In this paper we present a Bayesian hidden Markov model (HMM) which uses a non-parametric prior to infer a latent tagging for a sequence of words. HMMs have been popular for unsupervised PoS induction from its very beginnings (Brown et al., 1992), and justifiably so, as the most discriminating feature for deciding a word's PoS is its local syntactic context.

Our work brings together several strands of research including Bayesian non-parametric HMMs (Goldwater and Griffiths, 2007), Pitman-Yor language models (Teh, 2006b; Goldwater et al., 2006b), tagging constraints over word types (Brown et al., 1992) and the incorporation of morphological features (Clark, 2003). The result is a non-parametric Bayesian HMM which avoids overfitting, contains no free parameters, and exhibits good scaling properties. Our model uses a hierarchical Pitman-Yor process (PYP) prior to affect sophisicated smoothing over the transition and emission distributions. This allows the modelling of sub-word structure, thereby capturing tag-specific morphological variation. Unlike many existing approaches, our model is a principled generative model and does not include any hand tuned language specific features.

Inspired by previous successful approaches (Brown et al., 1992), we develop a new type-level inference procedure in the form of an MCMC sampler with an approximate method for incorporating the complex dependencies that arise between jointly sampled events. Our experimental evaluation demonstrates that our model, particularly when restricted to a single tag per type, produces

Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics, pages 865–874,
Portland, Oregon, June 19-24, 2011. ©2011 Association for Computational Linguistics

state-of-the-art results across a range of corpora and languages.

2 Background

Past research in unsupervised PoS induction has largely been driven by two different motivations: a task based perspective which has focussed on inducing word classes to improve various applications, and a linguistic perspective where the aim is to induce classes which correspond closely to annotated part-of-speech corpora. Early work was firmly situtated in the task-based setting of improving generalisation in language models. Brown et al. (1992) presented a simple first-order HMM which restricted word types to always be generated from the same class. Though PoS induction was not their aim, this restriction is largely validated by empirical analysis of treebanked data, and moreover conveys the significant advantage that all the tags for a given word type can be updated at the same time, allowing very efficient inference using the exchange algorithm. This model has been popular for language modelling and bilingual word alignment, and an implementation with improved inference called `mkcls` (Och, 1999)[1] has become a standard part of statistical machine translation systems.

The HMM ignores orthographic information, which is often highly indicative of a word's part-of-speech, particularly so in morphologically rich languages. For this reason Clark (2003) extended Brown et al. (1992)'s HMM by incorporating a character language model, allowing the modelling of limited morphology. Our work draws from these models, in that we develop a HMM with a one class per tag restriction and include a character level language model. In contrast to these previous works which use the maximum likelihood estimate, we develop a Bayesian model with a rich prior for smoothing the parameter estimates, allowing us to move to a trigram model.

A number of researchers have investigated a semi-supervised PoS induction task in which a tag dictionary or similar data is supplied a priori (Smith and Eisner, 2005; Haghighi and Klein, 2006; Goldwater and Griffiths, 2007; Toutanova and Johnson, 2008; Ravi and Knight, 2009). These systems achieve much higher accuracy than fully unsupervised systems, though it is unclear whether the tag dictionary assumption has real world application. We focus solely on the fully unsupervised scenario, which we believe is more practical for text processing in new languages and domains.

Recent work on unsupervised PoS induction has focussed on encouraging sparsity in the emission distributions in order to match empirical distributions derived from treebank data (Goldwater and Griffiths, 2007; Johnson, 2007; Gao and Johnson, 2008). These authors took a Bayesian approach using a Dirichlet prior to encourage sparse distributions over the word types emitted from each tag. Conversely, Ganchev et al. (2010) developed a technique to optimize the more desirable reverse property of the word types having a sparse posterior distribution over tags. Recently Lee et al. (2010) combined the one class per word type constraint (Brown et al., 1992) in a HMM with a Dirichlet prior to achieve both forms of sparsity. However this work approximated the derivation of the Gibbs sampler (omitting the interdependence between events when sampling from a collapsed model), resulting in a model which underperformed Brown et al. (1992)'s one-class HMM.

Our work also seeks to enforce both forms of sparsity, by developing an algorithm for type-level inference under the one class constraint. This work differs from previous Bayesian models in that we explicitly model a complex backoff path using a hierachical prior, such that our model jointly infers distributions over tag trigrams, bigrams and unigrams and whole words and their character level representation. This smoothing is critical to ensure adequate generalisation from small data samples.

Research in language modelling (Teh, 2006b; Goldwater et al., 2006a) and parsing (Cohn et al., 2010) has shown that models employing Pitman-Yor priors can significantly outperform the more frequently used Dirichlet priors, especially where complex hierarchical relationships exist between latent variables. In this work we apply these advances to unsupervised PoS tagging, developing a HMM smoothed using a Pitman-Yor process prior.

[1] Available from `http://fjoch.com/mkcls.html`.

3 The PYP-HMM

We develop a trigram hidden Markov model which models the joint probability of a sequence of latent tags, $\mathbf{t}$, and words, $\mathbf{w}$, as

$$P_\theta(\mathbf{t}, \mathbf{w}) = \prod_{l=1}^{L+1} P_\theta(t_l | t_{l-1}, t_{l-2}) P_\theta(w_l | t_l) \,,$$

where $L = |\mathbf{w}| = |\mathbf{t}|$ and $t_0 = t_{-1} = t_{L+1} = \$$ are assigned a sentinel value to denote the start or end of the sentence. A key decision in formulating such a model is the smoothing of the tag trigram and emission distributions, which would otherwise be too difficult to estimate from small datasets. Prior work in unsupervised PoS induction has employed simple smoothing techniques, such as additive smoothing or Dirichlet priors (Goldwater and Griffiths, 2007; Johnson, 2007), however this body of work has overlooked recent advances in smoothing methods used for language modelling (Teh, 2006b; Goldwater et al., 2006b). Here we build upon previous work by developing a PoS induction model smoothed with a sophisticated non-parametric prior. Our model uses a hierarchical Pitman-Yor process prior for both the transition and emission distributions, encoding a backoff path from complex distributions to successsively simpler ones. The use of complex distributions (e.g., over tag trigrams) allows for rich expressivity when sufficient evidence is available, while the hierarchy affords a means of backing off to simpler and more easily estimated distributions otherwise. The PYP has been shown to generate distributions particularly well suited to modelling language (Teh, 2006a; Goldwater et al., 2006b), and has been shown to be a generalisation of Kneser-Ney smoothing, widely recognised as the best smoothing method for language modelling (Chen and Goodman, 1996).

The model is depicted in the plate diagram in Figure 1. At its centre is a standard trigram HMM, which generates a sequence of tags and words,

$$\begin{aligned} t_l | t_{l-1}, t_{l-2}, T &\sim T_{t_{l-1}, t_{l-2}} \\ w_l | t_l, E &\sim E_{t_l} \,. \end{aligned}$$

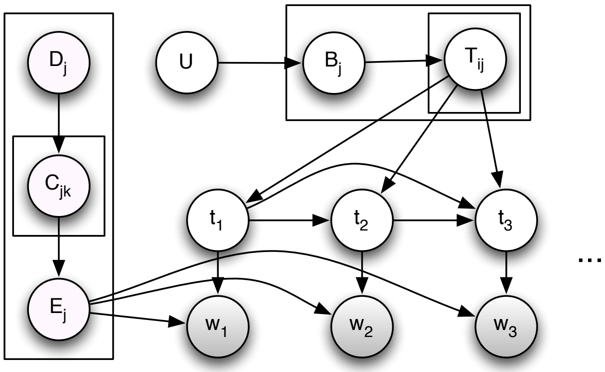

Figure 1: Plate diagram representation of the trigram HMM. The indexes i and j range over the set of tags and k ranges over the set of characters. Hyper-parameters have been omitted from the figure for clarity.

The trigram transition distribution, T_{ij}, is drawn from a hierarchical PYP prior which backs off to a bigram B_j and then a unigram U distribution,

$$\begin{aligned} T_{ij} | a^T, b^T, B_j &\sim \text{PYP}(a^T, b^T, B_j) \\ B_j | a^B, b^B, U &\sim \text{PYP}(a^B, b^B, U) \\ U | a^U, b^U &\sim \text{PYP}(a^U, b^U, \text{Uniform}) \,, \end{aligned}$$

where the prior over U has as its base distribition a uniform distribution over the set of tags, while the priors for B_j and T_{ij} back off by discarding an item of context. This allows the modelling of trigram tag sequences, while smoothing these estimates with their corresponding bigram and unigram distributions. The degree of smoothing is regulated by the hyper-parameters a and b which are tied across each length of n-gram; these hyper-parameters are inferred during training, as described in 3.1.

The tag-specific emission distributions, E_j, are also drawn from a PYP prior,

$$E_j | a^E, b^E, C \sim \text{PYP}(a^E, b^E, C_j) \,.$$

We consider two different settings for the base distribution C_j: 1) a simple uniform distribution over the vocabulary (denoted HMM for the experiments in section 4); and 2) a character-level language model (denoted HMM+LM). In many languages morphological regularities correlate strongly with a word's part-of-speech (e.g., suffixes in English), which we hope to capture using a basic character language model. This model was inspired by Clark (2003)

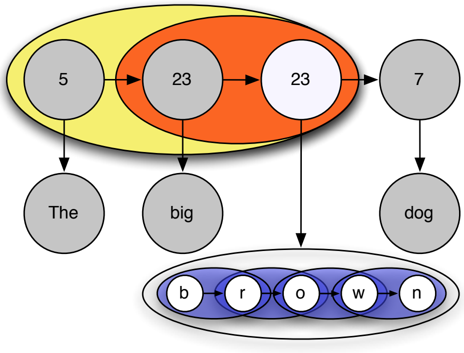

Figure 2: The conditioning structure of the hierarchical PYP with an embedded character language models.

who applied a character level distribution to the single class HMM (Brown et al., 1992). We formulate the character-level language model as a bigram model over the character sequence comprising word w_l,

$$\begin{aligned} w_{lk} | w_{lk-1}, t_l, C &\sim C_{t_l w_{lk-1}} \\ C_{jk} | a^C, b^C, D_j &\sim \mathrm{PYP}(a^C, b^C, D_j) \\ D_j | a^D, b^D &\sim \mathrm{PYP}(a^D, b^D, \mathrm{Uniform}) \,, \end{aligned}$$

where k indexes the characters in the word and, in a slight abuse of notation, the character itself, w_0 and is set to a special sentinel value denoting the start of the sentence (ditto for a final end of sentence marker) and the uniform base distribution ranges over the set of characters. We expect that the HMM+LM model will outperform the uniform HMM as it can capture many consistent morphological affixes and thereby better distinguish between different parts-of-speech. The HMM+LM is shown in Figure 2, illustrating the decomposition of the tag sequence into n-grams and a word into its component character bigrams.

3.1 Training

In order to induce a tagging under this model we use Gibbs sampling, a Markov chain Monte Carlo (MCMC) technique for drawing samples from the posterior distribution over the tag sequences given observed word sequences. We present two different sampling strategies: First, a simple Gibbs sampler which randomly samples an update to a single tag given all other tags; and second, a type-level sampler which updates all tags for a given word under a one-tag-per-word-type constraint. In order to extract a single tag sequence to test our model against the gold standard we find the tag at each site with maximum marginal probability in the sample set.

Following standard practice, we perform inference using a collapsed sampler whereby the model parameters U, B, T, E and C are marginalised out. After marginalisation the posterior distribution under a PYP prior is described by a variant of the Chinese Restaurant Process (CRP). The CRP is based around the analogy of a restaurant with an infinite number of tables, with customers entering one at a time and seating themselves at a table. The choice of table is governed by

$$P(z_l = k | \mathbf{z}_{-l}) = \begin{cases} \frac{n_k^- - a}{l-1+b} & 1 \leq k \leq K^- \\ \frac{K^- a + b}{l-1+b} & k = K^- + 1 \end{cases} \qquad (1)$$

where z_l is the table chosen by the lth customer, $\mathbf{z}_{-l}$ is the seating arrangement of the $l-1$ previous customers, n_k^- is the number of customers in $\mathbf{z}_{-l}$ who are seated at table k, $K^- = K(\mathbf{z}_{-l})$ is the total number of tables in $\mathbf{z}_{-l}$, and $z_1 = 1$ by definition. The arrangement of customers at tables defines a clustering which exhibits a power-law behavior controlled by the hyperparameters a and b.

To complete the restaurant analogy, a dish is then served to each table which is shared by all the customers seated there. This corresponds to a draw from the base distribution, which in our case ranges over tags for the transition distribution, and words for the observation distribution. Overall the PYP leads to a distribution of the form

$$P^T(t_l = i | \mathbf{z}_{-l}, \mathbf{t}_{-l}) = \frac{1}{n_{\mathbf{h}}^- + b^T} \times \qquad (2)$$
$$\left(n_{\mathbf{h}i}^- - K_{\mathbf{h}i}^- a^T + \left(K_{\mathbf{h}}^- a^T + b^T \right) P^B(i | \mathbf{z}_{-l}, \mathbf{t}_{-l}) \right),$$

illustrating the trigram transition distribution, where $\mathbf{t}_{-l}$ are all previous tags, $\mathbf{h} = (t_{l-2}, t_{l-1})$ is the conditioning bigram, $n_{\mathbf{h}i}^-$ is the count of the trigram $\mathbf{h}i$ in $\mathbf{t}_{-l}$, $n_{\mathbf{h}}^-$ the total count over all trigrams beginning with $\mathbf{h}$, $K_{\mathbf{h}i}^-$ the number of tables served dish i and $P^B(\cdot)$ is the base distribution, in this case the bigram distribution.

A hierarchy of PYPs can be formed by making the base distribution of a PYP another PYP, following a semantics whereby whenever a customer sits at an empty table in a restaurant, a new customer is also said to enter the restaurant for its base distribution. That is, each table at one level is equivalent to a customer at the next deeper level, creating the invariants: $K^{-}_{\mathbf{h}i} = n^{-}_{\mathbf{u}i}$ and $K^{-}_{\mathbf{u}i} = n^{-}_{i}$, where $\mathbf{u} = t_{l-1}$ indicates the unigram backoff context of **h**. The recursion terminates at the lowest level where the base distribution is static. The hierarchical setting allows for the modelling of elaborate backoff paths from rich and complex structure to successively simpler structures.

Gibbs samplers Both our Gibbs samplers perform the same calculation of conditional tag distributions, and involve first decrementing all trigrams and emissions affected by a sampling action, and then reintroducing the trigrams one at a time, conditioning their probabilities on the updated counts and table configurations as we progress.

The first local Gibbs sampler (PYP-HMM) updates a single tag assignment at a time, in a similar fashion to Goldwater and Griffiths (2007). Changing one tag affects three trigrams, with posterior

$$P(t_l|\mathbf{z}_{-l}, \mathbf{t}_{-l}, \mathbf{w}) \propto P(\mathbf{t}_{l\pm2}, w_l|\mathbf{z}_{-l\pm2}, \mathbf{t}_{-l\pm2}) \,,$$

where $l\pm2$ denotes the range $l-2, l-1, l, l+1, l+2$. The joint distribution over the three trigrams contained in $\mathbf{t}_{l\pm2}$ can be calculated using the PYP formulation. This calculation is complicated by the fact that these events are not independent; the counts of one trigram can affect the probability of later ones, and moreover, the table assignment for the trigram may also affect the bigram and unigram counts, of particular import when the same tag occurs twice in a row such as in Figure 2.

Many HMMs used for inducing word classes for language modelling include the restriction that all occurrences of a word type always appear with the same class throughout the corpus (Brown et al., 1992; Och, 1999; Clark, 2003). Our second sampler (PYP-1HMM) restricts inference to taggings which adhere to this one tag per type restriction. This restriction permits efficient inference techniques in which all tags of all occurrences of a word type are updated in parallel. Similar techniques have been used for models with Dirichlet priors (Liang et al., 2010), though one must be careful to manage the dependencies between multiple draws from the posterior.

The dependency on table counts in the conditional distributions complicates the process of drawing samples for both our models. In the non-hierarchical model (Goldwater and Griffiths, 2007) these dependencies can easily be accounted for by incrementing customer counts when such a dependence occurs. In our model we would need to sum over all possible table assignments that result in the same tagging, at all levels in the hierarchy: tag trigrams, bigrams and unigrams; and also words, character bigrams and character unigrams. To avoid this rather onerous marginalisation[2] we instead use expected table counts to calculate the conditional distributions for sampling. Unfortunately we know of no efficient algorithm for calculating the expected table counts, so instead develop a novel approximation

$$\mathrm{E}_{n+1}\left[K_i\right] \approx \mathrm{E}_n\left[K_i\right] + \frac{(a^U \mathrm{E}_n\left[K\right] + b^U)P_0(i)}{(n - \mathrm{E}_n\left[K_i\right] b^U) + (a^U \mathrm{E}_n\left[K\right] + b^U)P_0(i)} \,, \quad (3)$$

where K_i is the number of tables for the tag unigram i of which there are $n + 1$ occurrences, $\mathrm{E}_n\left[\cdot\right]$ denotes an expectation after observing n items and $\mathrm{E}_n\left[K\right] = \sum_j \mathrm{E}_n\left[K_j\right]$. This formulation defines a simple recurrence starting with the first customer seated at a table, $\mathrm{E}_1\left[K_i\right] = 1$, and as each subsequent customer arrives we *fractionally* assign them to a new table based on their conditional probability of sitting alone. These fractional counts are then carried forward for subsequent customers.

This approximation is tight for small n, and therefore it should be effective in the case of the local Gibbs sampler where only three trigrams are being resampled. For the type based resampling where large numbers of n are involved (consider resampling *the*), this approximation can deviate from the actual value due to errors accumulated in the recursion. Figure 3 illustrates a simulation demonstrating that the approximation is a close match for small a and n but underestimates the true value for high a

[2] Marginalisation is intractable in general, i.e. for the 1HMM where many sites are sampled jointly.

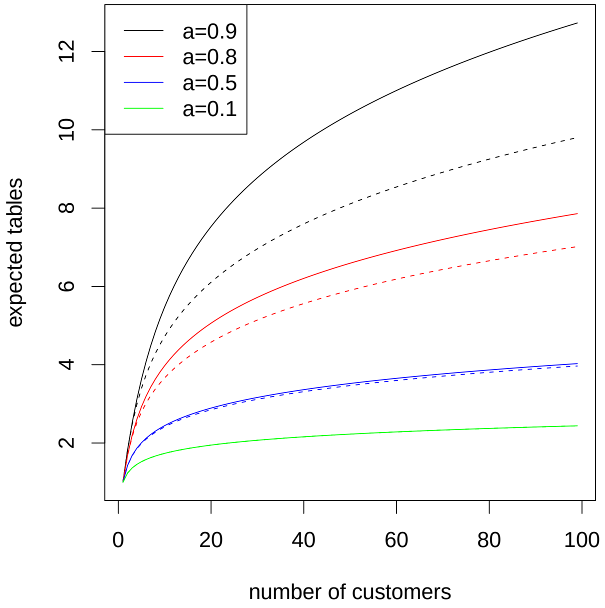

Figure 3: Simulation comparing the expected table count (solid lines) versus the approximation under Eq. 3 (dashed lines) for various values of a. This data was generated from a single PYP with $b = 1, P_0(i) = \frac{1}{4}$ and $n = 100$ customers which all share the same tag.

and n. The approximation was much less sensitive to the choice of b (not shown).

To resample a sequence of trigrams we start by removing their counts from the current restaurant configuration (resulting in $\mathbf{z}_{-}$). For each tag we simulate adding back the trigrams one at a time, calculating their probability under the given $\mathbf{z}_{-}$ plus the fractional table counts accumulated by Equation 3. We then calculate the expected table count contribution from this trigram and add it to the accumulated counts. The fractional table count from the trigram then results in a fractional customer entering the bigram restaurant, and so on down to unigrams. At each level we must update the expected counts before moving on to the next trigram. After performing this process for all trigrams under consideration and for all tags, we then normalise the resulting tag probabilities and sample an outcome. Once a tag has been sampled, we then add all the trigrams to the restaurants sampling their tables assignments explicitly (which are no longer fractional), recorded in $\mathbf{z}$. Because we do not marginalise out the table counts and our expectations are only approximate, this sampler will be biased. We leave to future work properly accounting for this bias, e.g., by devising a Metropolis Hastings acceptance test.

Sampling hyperparameters We treat the hyper-parameters $\{(a^x, b^x), x \in (U, B, T, E, C)\}$ as random variables in our model and infer their values. We place prior distributions on the PYP discount a^x and concentration b^x hyperparamters and sample their values using a slice sampler. For the discount parameters we employ a uniform Beta distribution ($a^x \sim \text{Beta}(1, 1)$), and for the concentration parameters we use a vague gamma prior ($b^x \sim \text{Gamma}(10, 0.1)$). All the hyper-parameters are resampled after every 5th sample of the corpus.

The result of this hyperparameter inference is that there are no user tunable parameters in the model, an important feature that we believe helps explain its consistently high performance across test settings.

4 Experiments

We perform experiments with a range of corpora to both investigate the properties of our proposed models and inference algorithms, as well as to establish their robustness across languages and domains. For our core English experiments we report results on the entire Penn. Treebank (Marcus et al., 1993), while for other languages we use the corpora made available for the CoNLL-X Shared Task (Buchholz and Marsi, 2006). We report results using the many-to-one (M-1) and v-measure (VM) metrics considered best by the evaluation of Christodoulopoulos et al. (2010). M-1 measures the accuracy of the model after mapping each predicted class to its most frequent corresponding tag, while VM is a variant of the F-measure which uses conditional entropy analogies of precision and recall. The log-posterior for the HMM sampler levels off after a few hundred samples, so we report results after five hundred. The 1HMM sampler converges more quickly so we use two hundred samples for these models. All reported results are the mean of three sampling runs.

An important detail for any unsupervised learning algorithm is its initialisation. We used slightly different initialisation for each of our inference strategies. For the unrestricted HMM we randomly assigned each word token to a class. For the restricted 1HMM we use a similar initialiser to

Model	**M-1**	**VM**
Prototype meta-model (CGS10)	76.1	68.8
MEMM (BBDK10)	75.5	-
`mkcls` (Och, 1999)	73.7	65.6
MLE 1HMM-LM (Clark, 2003)*	71.2	65.5
BHMM (GG07)	63.2	56.2
PR (Ganchev et al., 2010)*	62.5	54.8
Trigram PYP-HMM	69.8	62.6
Trigram PYP-1HMM	76.0	68.0
Trigram PYP-1HMM-LM	**77.5**	69.7
Bigram PYP-HMM	66.9	59.2
Bigram PYP-1HMM	72.9	65.9
Trigram DP-HMM	68.1	60.0
Trigram DP-1HMM	76.0	68.0
Trigram DP-1HMM-LM	76.8	**69.8**

Table 1: WSJ performance comparing previous work to our own model. The columns display the many-to-1 accuracy and the V measure, both averaged over 5 independent runs. Our model was run with the local sampler (HMM), the type-level sampler (1HMM) and also with the character LM (1HMM-LM). Also shown are results using Dirichlet Process (DP) priors by fixing $\mathbf{a} = 0$. The system abbreviations are CGS10 (Christodoulopoulos et al., 2010), BBDK10 (Berg-Kirkpatrick et al., 2010) and GG07 (Goldwater and Griffiths, 2007). Starred entries denote results reported in CGS10.

Clark (2003), assigning each of the k most frequent word types to its own class, and then randomly dividing the rest of the types between the classes.

As a baseline we report the performance of `mkcls` (Och, 1999) on all test corpora. This model seems not to have been evaluated in prior work on unsupervised PoS tagging, which is surprising given its consistently good performance.

First we present our results on the most frequently reported evaluation, the WSJ sections of the Penn. Treebank, along with a number of state-of-the-art results previously reported (Table 1). All of these models are allowed 45 tags, the same number of tags as in the gold-standard. The performance of our models is strong, particularly the 1HMM. We also see that incorporating a character language model (1HMM-LM) leads to further gains in performance, improving over the best reported scores under both M-1 and VM. We have omitted the results for the HMM-LM as experimentation showed that the local Gibbs sampler became hopelessly stuck, failing to mix due to the model's deep structure (its peak performance was $\approx 55\%$).

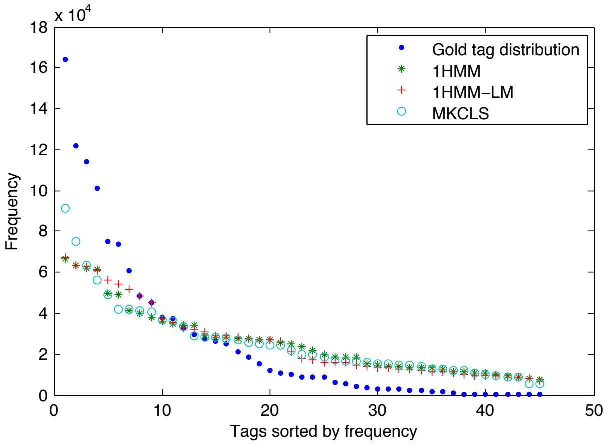

Figure 4: Sorted frequency of tags for WSJ. The gold standard distribution follows a steep exponential curve while the induced model distributions are more uniform.

To evaluate the effectiveness of the PYP prior we include results using a Dirichlet Process prior (DP). We see that for all models the use of the PYP provides some gain for the HMM, but diminishes for the 1HMM. This is perhaps a consequence of the expected table count approximation for the type-sampled PYP-1HMM: the DP relies less on the table counts than the PYP.

If we restrict the model to bigrams we see a considerable drop in performance. Note that the bigram PYP-HMM outperforms the closely related BHMM (the main difference being that we smooth tag bigrams with unigrams). It is also interesting to compare the bigram PYP-1HMM to the closely related model of Lee et al. (2010). That model incorrectly assumed independence of the conditional sampling distributions, resulting in a accuracy of 66.4%, well below that of our model.

Figures 4 and 5 provide insight into the behavior of the sampling algorithms. The former shows that both our models and `mkcls` induce a more uniform distribution over tags than specified by the treebank. It is unclear whether it is desirable for models to exhibit behavior closer to the treebank, which dedicates separate tags to very infrequent phenomena while lumping the large range of noun types into a single category. The graph in Figure 5 shows that the type-based 1HMM sampler finds a good tagging extremely quickly and then sticks with it,

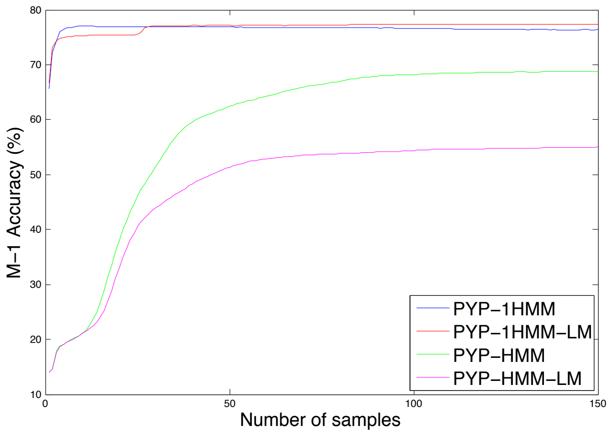

Figure 5: M-1 accuracy vs. number of samples.

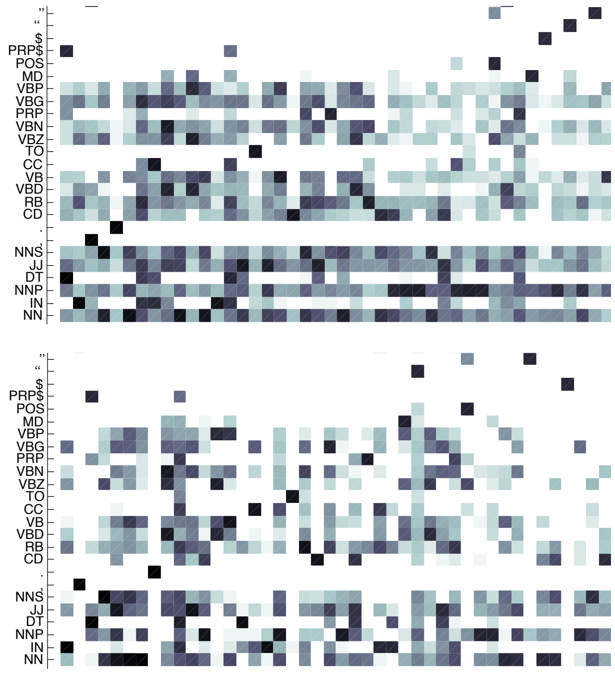

Figure 6: Cooccurence between frequent gold (y-axis) and predicted (x-axis) tags, comparing mkcls (top) and PYP-1HMM-LM (bottom). Both axes are sorted in terms of frequency. Darker shades indicate more frequent cooccurence and columns represent the induced tags.

save for the occasional step change demonstrated by the 1HMM-LM line. The locally sampled model is far slower to converge, rising slowly and plateauing well below the other models.

In Figure 6 we compare the distributions over WSJ tags for mkcls and the PYP-1HMM-LM. On the macro scale we can see that our model induces a sparser distribution. With closer inspection we can identify particular improvements our model makes. In the first column for mkcls and the third column for our model we can see similar classes with significant counts for DTs and PRPs, indicating a class that the models may be using to represent the start of sentences (informed by start transitions or capitalisation). This column exemplifies the sparsity of the PYP model's posterior.

We continue our evaluation on the CoNLL multilingual corpora (Table 2). These results show a highly consistent story of performance for our models across diverse corpora. In all cases the PYP-1HMM outperforms the PYP-HMM, which are both outperformed by the PYP-1HMM-LM. The character language model provides large gains in performance on a number of corpora, in particular those with rich morphology (Arabic +5%, Portuguese +5%, Spanish +4%). We again note the strong performance of the mkcls model, significantly beating recently published state-of-the-art results for both Dutch and Swedish. Overall our best model (PYP-1HMM-LM) outperforms both the state-of-the-art, where previous work exists, as well as mkcls consistently across all languages.

5 Discussion

The hidden Markov model, originally developed by Brown et al. (1992), continues to be an effective modelling structure for PoS induction. We have combined hierarchical Bayesian priors with a trigram HMM and character language model to produce a model with consistently state-of-the-art performance across corpora in ten languages. However our analysis indicates that there is still room for improvement, particularly in model formulation and developing effective inference algorithms.

Induced tags have already proven their usefulness in applications such as Machine Translation, thus it will prove interesting as to whether the improvements seen from our models can lead to gains in downstream tasks. The continued successes of models combining hierarchical Pitman-Yor priors with expressive graphical models attests to this framework's enduring attraction, we foresee continued interest in applying this technique to other NLP tasks.

Language	**mkcls**	**HMM**	**1HMM**	**1HMM-LM**	**Best pub.**	**Tokens**	**Tag types**
Arabic	58.5	57.1	62.7	**67.5**	-	54,379	20
Bulgarian	66.8	67.8	69.7	**73.2**	-	190,217	54
Czech	59.6	62.0	66.3	**70.1**	-	1,249,408	12[c]
Danish	62.7	69.9	73.9	**76.2**	66.7[⋆]	94,386	25
Dutch	64.3	66.6	68.7	**70.4**	67.3[†]	195,069	13[c]
Hungarian	54.3	65.9	69.0	**73.0**	-	131,799	43
Portuguese	68.5	72.1	73.5	**78.5**	75.3[⋆]	206,678	22
Spanish	63.8	71.6	74.7	**78.8**	73.2[⋆]	89,334	47
Swedish	64.3	66.6	67.0	**68.6**	60.6[†]	191,467	41

Table 2: Many-to-1 accuracy across a range of languages, comparing our model with `mkcls` and the best published result ([⋆]Berg-Kirkpatrick et al. (2010) and [†]Lee et al. (2010)). This data was taken from the CoNLL-X shared task training sets, resulting in listed corpus sizes. Fine PoS tags were used for evaluation except for items marked with [c], which used the coarse tags. For each language the systems were trained to produce the same number of tags as the gold standard.

References

Taylor Berg-Kirkpatrick, Alexandre Bouchard-Côté, John DeNero, and Dan Klein. 2010. Painless unsupervised learning with features. In *Human Language Technologies: The 2010 Annual Conference of the North American Chapter of the Association for Computational Linguistics*, pages 582–590, Los Angeles, California, June. Association for Computational Linguistics.

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

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, Morristown, NJ, USA. Association for Computational Linguistics.

Stanley F. Chen and Joshua Goodman. 1996. An empirical study of smoothing techniques for language modeling. In *Proceedings of the 34th annual meeting on Association for Computational Linguistics*, pages 310–318, Morristown, NJ, 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.

Alexander Clark. 2003. Combining distributional and morphological information for part of speech induction. In *Proceedings of the tenth Annual Meeting of the European Association for Computational Linguistics (EACL)*, pages 59–66.

Trevor Cohn, Phil Blunsom, and Sharon Goldwater. 2010. Inducing tree-substitution grammars. *Journal of Machine Learning Research*, pages 3053–3096.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2010. Posterior regularization for structured latent variable models. *Journal of Machine Learning Research*, 99:2001–2049, August.

Jianfeng Gao and Mark Johnson. 2008. A comparison of bayesian estimators for unsupervised hidden markov model pos taggers. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08, pages 344–352, Morristown, NJ, USA. Association for Computational Linguistics.

Sharon Goldwater and Tom Griffiths. 2007. A fully bayesian approach to unsupervised part-of-speech tagging. In *Proc. of the 45th Annual Meeting of the ACL (ACL-2007)*, pages 744–751, Prague, Czech Republic, June.

Sharon Goldwater, Tom Griffiths, and Mark Johnson. 2006a. Contextual dependencies in unsupervised word segmentation. In *Proc. of the 44th Annual Meeting of the ACL and 21st International Conference on Computational Linguistics (COLING/ACL-2006)*, Sydney.

Sharon Goldwater, Tom Griffiths, and Mark Johnson. 2006b. Interpolating between types and tokens by estimating power-law generators. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural*

Information Processing Systems 18, pages 459–466. MIT Press, Cambridge, MA.

Aria Haghighi and Dan Klein. 2006. Prototype-driven learning for sequence models. In *Proceedings of the main conference on Human Language Technology Conference of the North American Chapter of the Association of Computational Linguistics*, pages 320–327, Morristown, NJ, USA. Association for Computational Linguistics.

Mark Johnson. 2007. Why doesnt EM find good HMM POS-taggers? In *Proc. of the 2007 Conference on Empirical Methods in Natural Language Processing (EMNLP-2007)*, pages 296–305, Prague, Czech Republic.

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, Morristown, NJ, USA. Association for Computational Linguistics.

P. Liang, M. I. Jordan, and D. Klein. 2010. Type-based MCMC. In *North American Association for Computational Linguistics (NAACL)*.

Mitchell P. Marcus, Mary Ann Marcinkiewicz, and Beatrice Santorini. 1993. Building a large annotated corpus of English: the Penn treebank. *Computational Linguistics*, 19(2):313–330.

Franz Josef Och. 1999. An efficient method for determining bilingual word classes. In *Proceedings of the ninth conference on European chapter of the Association for Computational Linguistics*, pages 71–76, Morristown, NJ, USA. Association for Computational Linguistics.

Sujith Ravi and Kevin Knight. 2009. Minimized models for unsupervised part-of-speech tagging. In *Proceedings of the Joint Conferenceof the 47th Annual Meeting of the Association for Computational Linguistics and the 4th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing (ACL-IJCNLP)*, pages 504–512.

Noah A. Smith and Jason Eisner. 2005. Contrastive estimation: Training log-linear models on unlabeled data. In *Proceedings of the 43rd Annual Meeting of the Association for Computational Linguistics (ACL)*, pages 354–362, Ann Arbor, Michigan, June.

Y. W. Teh. 2006a. A hierarchical Bayesian language model based on Pitman-Yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 985–992.

Yee Whye Teh. 2006b. A hierarchical bayesian language model based on pitman-yor processes. In *Proceedings of the 21st International Conference on Computational Linguistics and the 44th annual meeting of the Association for Computational Linguistics*, ACL-44, pages 985–992, Morristown, NJ, USA. Association for Computational Linguistics.

Kristina Toutanova and Mark Johnson. 2008. A bayesian lda-based model for semi-supervised part-of-speech tagging. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1521–1528. MIT Press, Cambridge, MA.