Two Decades of Unsupervised POS tagging---How Far Have We Come?

Two Decades of Unsupervised POS induction: How far have we come?

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

Part-of-speech (POS) induction is one of the most popular tasks in research on unsupervised NLP. Many different methods have been proposed, yet comparisons are difficult to make since there is little consensus on evaluation framework, and many papers evaluate against only one or two competitor systems. Here we evaluate seven different POS induction systems spanning nearly 20 years of work, using a variety of measures. We show that some of the oldest (and simplest) systems stand up surprisingly well against more recent approaches. Since most of these systems were developed and tested using data from the WSJ corpus, we compare their generalization abilities by testing on both WSJ and the multilingual Multext-East corpus. Finally, we introduce the idea of evaluating systems based on their ability to produce cluster prototypes that are useful as input to a prototype-driven learner. In most cases, the prototype-driven learner outperforms the unsupervised system used to initialize it, yielding state-of-the-art results on WSJ and improvements on non-English corpora.

1 Introduction

In recent years, unsupervised learning has become a hot area in NLP, in large part due to the use of sophisticated machine learning approaches which promise to deliver better results than more traditional methods. Often the new approaches are tested using part-of-speech (POS) tagging as an example application, and usually they are shown to perform better than one or another comparison system. However, it is difficult to draw overall conclusions about the relative performance of unsupervised POS tagging systems because of differences in evaluation measures, and the fact that no paper includes direct comparisons against more than a few other systems. In this paper, we attempt to remedy that situation by providing a comprehensive evaluation of seven different POS induction systems spanning nearly 20 years of research. We focus specifically on POS *induction* systems, where no prior knowledge is available, in contrast to POS *disambiguation* systems (Merialdo, 1994; Toutanova and Johnson, 2007; Naseem et al., 2009; Ravi and Knight, 2009; Smith and Eisner, 2005), which use a dictionary to provide possible tags for some or all of the words in the corpus, or *prototype-driven* systems (Haghighi and Klein, 2006), which use a small set of prototypes for each tag class, but no dictionary. Our motivation stems from another part of our own research, in which we are trying to use NLP systems on over 50 low-density languages (some of them dead) where both tagged corpora and language speakers are mostly unavailable. We therefore desire to use these systems straight out of the box and to know how well we can expect them to work.

One difficulty in evaluating POS induction systems is that there is no straightforward way to map the clusters found by the algorithm onto the gold standard tags; moreover, some systems are designed to induce the number of clusters as well as their contents, so the number of found clusters may not match either the gold standard or that of another system. Nevertheless, most recent papers have used mapping-based performance measures (either *one-to-one* or *many-to-one* accuracy). Here, we argue that the entropy-based V-Measure (Rosenberg and

Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, pages 575–584,
MIT, Massachusetts, USA, 9-11 October 2010.

Hirschberg, 2007) is more useful in many cases, being more stable across different numbers of found and true clusters, and avoiding several of the problems with another commonly used entropy-based measure, Variation of Information (Meilă, 2003).

Using V-Measure along with several other evaluation measures, we compare the performance of the different induction systems on both WSJ (the data on which most systems were developed and tested) and Multext East, a corpus of parallel texts in eight different languages. We find that for virtually all measures and datasets, older systems using relatively simple models and algorithms (Brown et al., 1992; Clark, 2003) work as well or better than systems using newer and often far more sophisticated and time-consuming machine learning methods (Goldwater and Griffiths, 2007; Johnson, 2007; Graca et al., 2009; Berg-Kirkpatrick et al., 2010). Thus, although these newer methods have introduced potentially useful machine learning techniques, they should not be assumed to provide the best performance for unsupervised POS induction.

In addition to our review and comparison, we introduce a new way to both evaluate and potentially improve a POS induction system. Our method is based on the prototype-driven learning system of Haghighi and Klein (2006), which achieves very good performance by using a hand-selected list of prototypes for each syntactic cluster. We instead use the existing POS induction systems to induce prototypes automatically, and evaluate the systems based on the quality of their prototypes. We find that the oldest system tested (Brown et al., 1992) produces the best prototypes, and that using these prototypes as input to Haghighi and Klein's system yields state-of-the-art performance on WSJ and improvements on seven of the eight non-English corpora.

2 POS Induction Systems

We describe each system only briefly; for details, see the respective papers, cited below. Each system outputs a set of syntactic clusters C; except where noted, the target number of clusters $|C|$ must be specified as an input parameter. Since we are interested in out-of-the-box performance, we use the default parameter settings for each system, except for $|C|$, which is varied in some of our experiments. The systems are as follows:[1]

[brown]: Class-based n-grams (Brown et al., 1992). This is the oldest and one of the simplest systems we tested. It uses a bigram model where each word type is assigned to a latent class (a hard assignment), and the probability of the corpus $w_1 \dots w_n$ is computed as $P(w_1|c_1) \prod_{i=2}^{n} P(w_i|c_i)P(c_i|c_{i-1})$, where c_i is the class of w_i. The goal is to optimize the probability of the corpus under this model. The authors use an approximate search procedure: greedy agglomerative hierarchical clustering followed by a step in which individual word types are considered for movement to a different class if this improves the corpus probability.

[clark]: Class-based n-grams with morphology (Clark, 2003). This system uses a similar model to the previous one, and also clusters word types (rather than tokens, as the rest of the systems do). The main differences between the systems are that clark uses a slightly different approximate search procedure, and that he augments the probabilistic model with a prior that prefers clusterings where morphologically similar words are clustered together. The morphology component is implemented as a single-order letter HMM.

[cw]: Chinese Whispers graph clustering (Biemann, 2006). Unlike the other systems we consider, this one induces the value of $|C|$ rather than taking it as an input parameter.[2] The system uses a graph clustering algorithm called *Chinese Whispers* that is based on contextual similarity. The algorithm works in two stages. The first clusters the most frequent 10,000 words (*target words*) based on their context statistics, with contexts formed from the most frequent 150-250 words (*feature words*) that appear ei-

[1] Implementations were obtained from:
brown: `http://www.cs.berkeley.edu/~pliang/software/brown-cluster-1.2.zip` (Percy Liang),
clark: `http://www.cs.rhul.ac.uk/home/alexc/pos2.tar.gz` (Alex Clark),
cw: `http://wortschatz.uni-leipzig.de/%7Ecbiemann/software/jUnsupos1.0.zip` (Chris Biemann),
bhmm, **vbhmm**, **pr**, **feat**: by request from the authors of the respective papers.

[2] Another recent model that induces $|C|$ is the Infinite HMM (Van Gael et al., 2009). Unfortunately, we were unable to obtain code for the IHMM in time to include it in our analysis. Van Gael et al. (2009) report results of around 59% V-Measure on WSJ, with 194 induced clusters, which is not as good as the best system scores in Section 4.

ther to the left or right of a target word. The second stage deals with medium and low frequency words and uses pairwise similarity scores calculated by the number of shared neighbors between two words in a 4-word context window. The final clustering is a combination of the clusters obtained in the two stages. While the number of target words, feature words, and window size are in principle parameters of the algorithm, they are hard-coded in the implementation we used and we did not change them.

[bhmm]: Bayesian HMM with Gibbs sampling (Goldwater and Griffiths, 2007). This system is based on a standard HMM for POS tagging. It differs from the standard model by placing Dirichlet priors over the multinomial parameters defining the state-state and state-emission distributions, and uses a collapsed Gibbs sampler to infer the hidden tags. The Dirichlet hyperparameters α (which controls the sparsity of the transition probabilities) and β (which controls the sparsity of the emission probabilities) can be fixed or inferred. We used a bigram version of this model with hyperparameter inference.

[vbhmm]: Bayesian HMM with variational Bayes (Johnson, 2007). This system uses the same bigram model as **bhmm**, but uses variational Bayesian EM for inference. We fixed the α and β parameters to 0.1, values that appeared to be reasonable based on Johnson (2007), and which were also used by Graca et al. (2009).

[pr]: Sparsity posterior-regularization HMM (Graca et al., 2009). The Bayesian approaches described above encourage sparse state-state and state-emission distributions only indirectly through the Dirichlet priors. This system, while utilizing the same bigram HMM, encourages sparsity directly by constraining the *posterior* distributions using the posterior regularization framework (Ganchev et al., 2009). A parameter σ controls the strengths of the constraints (default = 25). Following Graca et al. (2009), we set $\alpha = \beta = 0.1$.

[feat]: Feature-based HMM (Berg-Kirkpatrick et al., 2010). This system uses a model that has the structure of a standard HMM, but assumes that the state-state and state-emission distributions are logistic, rather than multinomial. The logistic distributions allow the model to incorporate local features of the sort often used in discriminative models. The default features are morphological, such as character trigrams and capitalization.

3 Evaluation Measures

One difficulty in comparing POS induction methods is in finding an appropriate evaluation measure. Many different measures have been proposed over the years, but there is still no consensus on which is best. In addition, some measures with supposed theoretical advantages, such as Variation of Information (VI) (Meilă, 2003) have had little empirical analysis. Our goal in this section is to determine which of these measures is most sensible for evaluating the systems presented above. We first describe each measure before presenting empirical results. Except for VI, all measures range from 0 to 1, with higher scores indicating better performance.

[many-to-1]: Many-to-one mapping accuracy (also known as *cluster purity*) maps each cluster to the gold standard tag that is most common for the words in that cluster (henceforth, the *preferred tag*), and then computes the proportion of words tagged correctly. More than one cluster may be mapped to the same gold standard tag. This is the most commonly used metric across the literature as it is intuitive and creates a meaningful POS sequence out of the cluster identifiers. However, it tends to yield higher scores as $|C|$ increases, making comparisons difficult when $|C|$ can vary.

[crossval]: Cross-validation accuracy (Gao and Johnson, 2008) is intended to address the problem with many-to-one accuracy which is that assigning each word to its own class yields a perfect score. In this measure, the first half of the corpus is used to obtain the many-to-one mapping of clusters to tags, and this mapping is used to compute the accuracy of the clustering on the second half of the corpus.

[1-to-1]: One-to-one mapping accuracy (Haghighi and Klein, 2006) constrains the mapping from clusters to tags, so that at most one cluster can be mapped to any tag. The mapping is performed greedily. In general, as the number of clusters increases, fewer clusters will be mapped to their preferred tag and scores will decrease (especially if the number of clusters is larger than the number of tags, so that some clusters are unassigned and receive zero credit). Again, this makes it difficult to

compare solutions with different values of $|C|$.

[vi]: Variation of Information (Meilă, 2003) is an information-theoretic measure that regards the system output C and the gold standard tags T as two separate clusterings, and evaluates the amount of information lost in going from C to T and the amount of information gained, i.e., the sum of the conditional entropy of each clustering conditioned on the other. More formally, $VI(C,T) = H(T|C) + H(C|T) = H(C) + H(T) - 2I(C,T)$, where $H(.)$ is the entropy function and $I(.)$ is the mutual information. VI and other entropy-based measures have been argued to be superior to accuracy-based measures such as those above, because they consider not only the majority tag in each cluster, but also whether the remainder of the cluster is more or less homogeneous. Unlike the other measures we consider, lower scores are better (since VI measures the difference between clusterings in bits).

[vm]: V-Measure (Rosenberg and Hirschberg, 2007) is another entropy-based measure that is designed to be analogous to F-measure, in that it is defined as the weighted harmonic mean of two values, homogeneity (h, the precision analogue) and completeness (c, the recall analogue):

$$h = 1 - \frac{H(T|C)}{H(T)} \quad (1)$$

$$c = 1 - \frac{H(C|T)}{H(C)} \quad (2)$$

$$VM = \frac{(1+\beta)hc}{(\beta h) + c} \quad (3)$$

As with F-measure, β is normally set to 1.

[vmb]: V-beta is an extension to V-Measure, proposed by (Vlachos et al., 2009). They noted that V-Measure favors clusterings where the number of clusters $|C|$ is larger than the number of POS tags $|T|$. To address this issue the parameter β in equation 3 is set to $|C|/|T|$ in order adjust the balance between homogeneity and completeness.

[s-fscore]: Substitutable F-score (Frank et al., 2009). One potential issue with all of the above measures is that they require a gold standard tagging to compute. This is normally available during development of a system, but if the system is deployed on a novel language a gold standard may not be available. In addition, there is the question of whether the gold standard itself is "correct". Recently, Frank et al. (2009) proposed this novel evaluation measure that requires no gold standard, instead using the concept of substitutability to evaluate performance. Instead of comparing the system's clusters C to gold standard clusters T, they are compared to a set of clusters S created from *substitutable frames*, i.e., clusters of words that occur in the same syntactic environment. Ideally a substitutable frame would be created by sentences differing in only one word (e.g. "I want the blue ball." and "I want the red ball.") and the resulting cluster would contain the words that change (e.g. [blue, red]). However since it is almost impossible to find these types of sentences in real-world corpora, the authors use frames created by two words appearing in the corpus with exactly one word between (e.g. the —- ball). Once the substitutable clusters have been created, they can be used to calculate the Precision (SP), Recall (SR) and F-score (SF) of the system's clustering:

$$SP = \frac{\sum_{s \in S} \sum_{c \in C} |s \cap c|(|s \cap c| - 1)}{\sum_{c \in C} |c|(|c| - 1)} \quad (4)$$

$$SR = \frac{\sum_{s \in S} \sum_{c \in C} |s \cap c|(|s \cap c| - 1)}{\sum_{s \in S} |s|(|s| - 1)} \quad (5)$$

$$SF = \frac{2 \cdot SP \cdot SR}{SP + SR} \quad (6)$$

3.1 Empirical results

We mentioned a few strengths and weaknesses of each evaluation method above; in this section we present some empirical results to expand on these claims. First, we examine the effects of varying $|C|$ on the behavior of the evaluation measures, while keeping the number of gold standard tags the same ($|T| = 45$). Results were obtained by training and evaluating each system on the full WSJ portion of the Penn Treebank corpus (Marcus et al., 1993). Figure 1 shows the results from the Brown system for $|C|$ ranging from 1 to 200; the same trends were observed for all other systems.[3] In addition, Table 1 provides results for the two extremes of $|C| = 1$ (**all** words assigned to the same cluster) and $|C|$ equal to the size of the corpus (a **single** word per cluster), as

[3]The results reported in this paper are only a fraction of the total from our experiments; given the number of parameters, models and measures tested, we obtained over 15000 results. The full set of results can be found at `http://homepages.inf.ed.ac.uk/s0787820/pos/`.

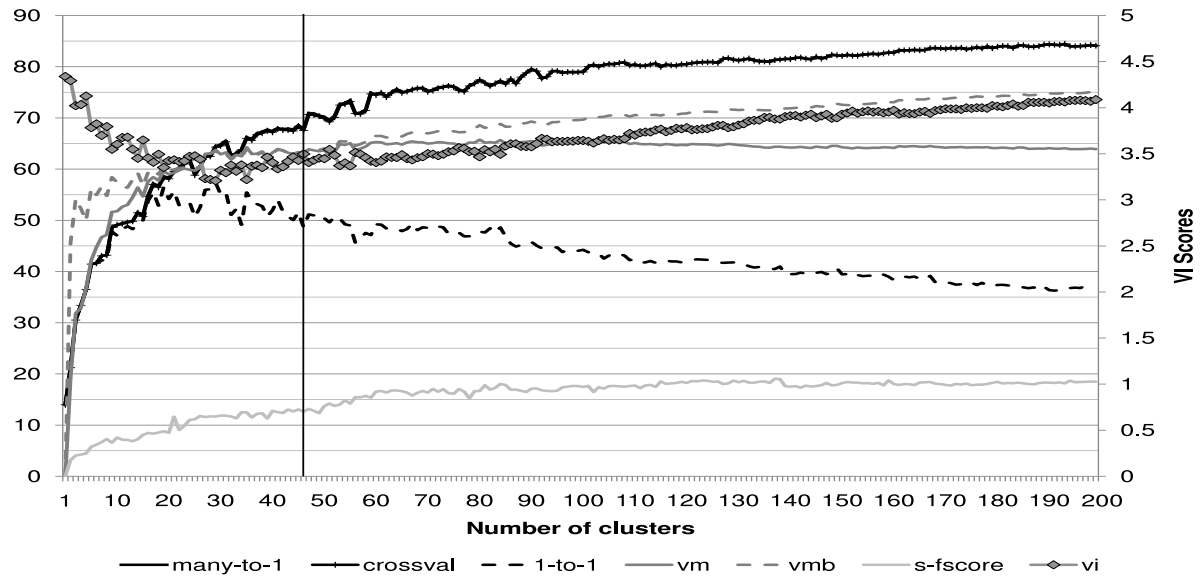

Figure 1: Scores for all evaluation measures as a function of the number of clusters returned [model:brown, corpus:wsj, $|C|$:{1-200}, $|T|$:45]. The right-hand y-axis shows VI scores (lower is better); the left-hand y-axis shows percentage scores for all other measures. The vertical line indicates $|T|$. Many-to-1 is invisible as it tracks crossval so closely.

measure	super	random	all	single
many-to-1	97.85	13.97	13.97	100
crossval	97.59	13.98	13.98	0
1-to-1	97.86	2.42	13.97	0.01
vi	0.35	9.81	4.33	15.82
vm	95.98	0.02	0	35.42
vmb	95.98	0	0	99.99
s-fscore	7.53	0.50	0	0

Table 1: Baseline scores for the different evaluation measures on the WSJ corpus. For all measures except VI higher is better.

well as two other baselines (a **super**vised tagging[4] and a **random** clustering with $|C| = 45$).

These empirical results confirm that certain measures favor solutions with many clusters, while others prefer fewer clusters. As expected, **many-to-1** correlates positively with $|C|$, rising to almost 85% with $|C| = 200$ and reaching 100% when the number of clusters is maximal (i.e., **single**). Recall that **crossval** was proposed as a possible solution to this problem, and it does solve the extreme case of **single**, yielding 0% accuracy rather than 100%. However, it patterns just like **many-to-1** for up to 200 clusters, suggesting that there is very little difference between the two for any reasonable number of clusters, and we should be wary of using either one when $|C|$ may vary.

In contrast to these measures are **1-to-1** and **vi**: for the most part, they yield worse performance (lower **1-to-1**, higher **vi**) as $|C|$ increases. However, in this case the trend is not monotonic: there is an initial improvement in performance before the decrease begins. One might hope that the peak in performance would occur when the number of clusters is approximately equal to the number of gold standard tags; however, the best performance for both **1-to-1** and **vi** occurs with approximately 25-30 clusters, many fewer than the gold standard 45.

Next we consider **vm** and **vmb**. Interestingly, although **vmb** was proposed as a way to correct for the supposed tendency of **vm** to increase with increasing $|C|$, we find that **vm** is actually more stable than **vmb** over different values of $|C|$. Thus, if the goal is to compare systems producing different numbers of clusters (especially important for systems that induce the number of clusters), then **vm** seems more appropriate than any of the above measures, which are more standard in the literature.

Finally, we analyze the behavior of the gold-standard-independent measure, **s-fscore**. On the positive side, this measure assigns scores of 0 to the

[4]We used the Stanford Tagger trained on the WSJ corpus: `http://nlp.stanford.edu/software/tagger.shtml`.

two extreme cases of **all** and **single**, and is relatively stable across different values of $|C|$ after an initial increase. It assigns a lower score to the supervised system than to **brown**, indicating that words in the supervised clusters (which are very close to the gold standard) are actually less substitutable than words in the unsupervised clusters. This is probably due to the fact that the gold standard encodes "pure" syntactic classes, while substitutability also depends on semantic characteristics (which tend to be picked up by unsupervised clustering systems as well). Another potential problem with this measure is that it has a very small dynamic range – while scores as high as 1 are theoretically possible, in practice they will never be achieved, and we see that the actual range of scores observed are all under 20%.

We conclude that there is probably no single evaluation measure that is best for all purposes. If a gold standard is available, then **many-to-1** and **1-to-1** are the most intuitive measures, but should not be used when $|C|$ is variable, and do not account for differences in the errors made. While **vi** has been popular as an entropy-based alternative to address the latter problem, its scores are not easy to interpret (being on a scale of bits) and it still has the problem of incomparability across different $|C|$. Overall, **vm** seems to be the best general-purpose measure that combines an entropy-based score with an intuitive 0-1 scale and stability over a wide range of $|C|$.

4 System comparison

Having provided some intuitions about the behavior of different evaluation methods, we move on to evaluating the various systems presented in Section 2. We first present results for the same WSJ corpus used above. However, because most of the systems were initially developed on this corpus, and often evaluated only on it, there is a question of whether their methods and/or hyperparameters are overly specific to the domain or to the English language. This is a particularly pertinent question since a primary argument in favor of unsupervised systems is that they are easier to port to a new language or domain than supervised systems. To address this question, we evaluate all the systems as well on the multilingual Multext East corpus (Erjavec, 2004), without changing any of the parameter settings. $|C|$ was set to 45 for all of the experiments reported in this section. Based on our assessment of evaluation measures above, we report VM scores as the most reliable measure across different systems and cluster set sizes; to facilitate comparisons with previous papers, we also report many-to-one and one-to-one accuracy.

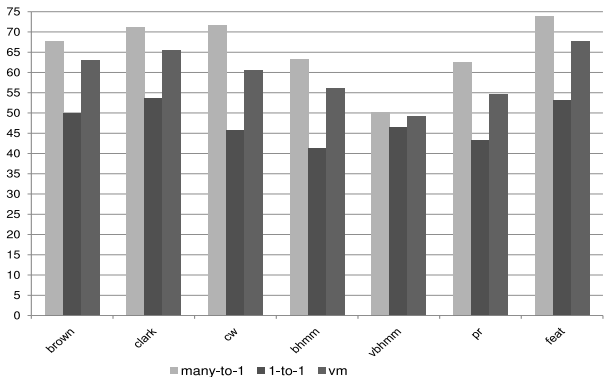

Figure 2: Performance of the different systems on WSJ, using three different measures [$|C|$:45, $|T|$:45]

system	runtime
brown	~10 min.
clark	~40 min.
cw	~10 min.
bhmm	~4 hrs.
vbhmm	~10 hrs.
pr	~10 hrs.*
feat	~40 hrs.*

Table 2: Runtimes for the different systems on WSJ [$|C|$:45]. ***pr** and **feat** have multithreading implementations and ran on 16 cores.

4.1 Results on WSJ

Figure 2 presents results for all seven systems, with approximate runtimes shown in Table 2. While these algorithms have not necessarily been optimized for speed, there is a fairly clear distinction between the older type-clustering models (**brown**, **clark**) and the graph-based algorithm (**cw**) on the one hand, and the newer machine-learning approaches (**bhmm**, **vbhmm**, **pr**, **feat**) on the other, with the former being much faster to run. Despite their faster runtimes and less sophisticated methods, however, these systems perform surprisingly well in comparison to the latter group. Even the oldest and perhaps simplest method (**brown**) outperforms the two BHMMs and posterior regularization on all measures. Only

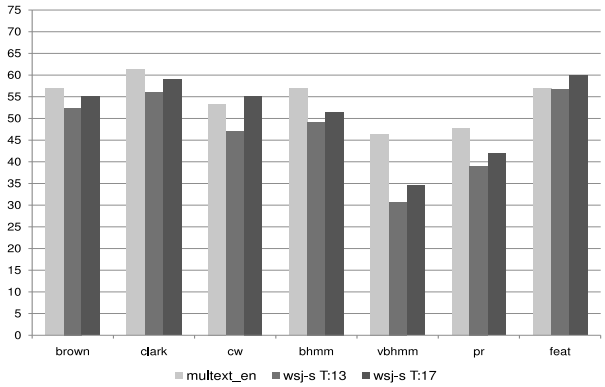

Figure 3: VM scores for the different systems on English Multext-East and WSJ-S corpora [$|C|$:45, $|T|$:{14,17}]

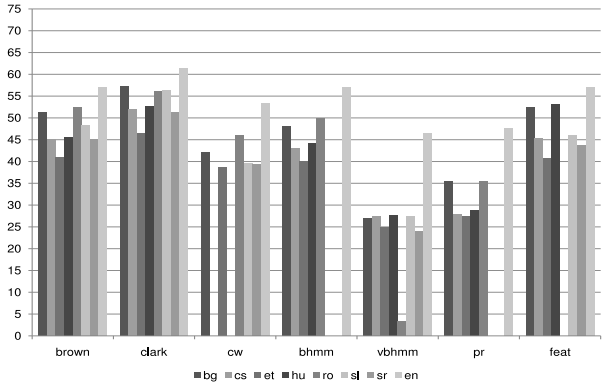

Figure 4: VM scores for the different systems on the eight Multext-East corpora [$|C|$:45, $|T|$:14]

the very latest approach (**feat**) rivals **clark**, showing slightly better performance on two of the three measures (**clark**: 71.2, 53.8, 65.5 on many-to-one, one-to-one, VM; **feat**: 73.9, 53.3, 67.7). The **cw** system returns a total of 568 clusters on this data set, so the many-to-one and one-to-one measures are not strictly comparable to the other systems; on VM this system achieves middling performance.

We note that the two best-performing systems, **clark** and **feat**, are also the only two to use morphological information. Since the clustering algorithms used by **brown** and **clark** are quite similar, the difference in performance between the two can probably be attributed to the extra information provided by the morphology. This supports the (unsurprising) conclusion that incorporating morphological features is generally helpful for POS induction.

4.2 Results on other corpora

We now examine whether either the relative or absolute performance of the different systems holds up when tested on a variety of different languages. For these experiments, we used the 1984 portion of the Multext-East corpus (~7k sentences), which contains parallel translations of Orwell's *1984* in 8 different languages: Bulgarian[**bg**], Czech[**cs**], Estonian[**et**], Hungarian[**hu**], Romanian[**ro**], Slovene[**sl**], Serbian[**sr**] and English[**en**]. We also included a 7k sentence version of the WSJ corpus [**wsj-s**] to help differentiate effects of corpus size from those of domain/language. For the WSJ corpora we experimented with two standardly used tagsets: the original PTB 45-tag gold standard and a coarser set of 17 tags previously used by several researchers working on unsupervised POS tagging (Smith and Eisner, 2005; Goldwater and Griffiths, 2007; Johnson, 2007). For the Multext-East corpus only a coarse 14-tag tagset was available.[5] Finally, to facilitate direct comparisons of genre while controlling for the size of both the corpus and the tag set, we also created a further collapsed 13-tag set for WSJ.[6]

Figure 3 illustrates the abilities of the different systems to generalize across different genres of English text. Comparing the results for the Multext-East English corpus and the small WSJ corpus with 13 tags (i.e., controlling as much as possible for corpus size and number of gold standard tags), we see that despite being developed on WSJ, the systems actually perform better on Multext-East. This is encouraging, since it suggests that the methods and hyperparameters of the algorithms are not strongly tied to WSJ. It also suggests that Multext-East is in some sense an easier corpus than WSJ. Indeed, the distribution of vocabulary items supports this view: the 100 most frequent words account for 48% of the WSJ corpus, but 57% of the 1984 novel. It is also worth pointing out that, although previous researchers have reduced the 45-tag WSJ set to 17 tags in order to create an easier task for unsupervised learning (and to decrease training time), reducing the tag set further to 13 tags actually decreases performance, since some distinctions found by the sys-

[5] Out of the 14 tags only 11 are shared across all languages. For details c.f. Appendix B in (Naseem et al., 2009).

[6] We tried to make the meanings of the tags as similar as possible between the two corpora; we had to create 13 rather than 14 WSJ tags for this reason. Our 13-tag set can be found at `http://homepages.inf.ed.ac.uk/s0787820/pos/`.

tems (e.g., between different types of punctuation) are collapsed in the gold standard.

Figure 4 gives the results of the different systems on the various languages.[7] Not surprisingly, all the algorithms perform best on English, often by a wide margin, suggesting that they are indeed tuned better towards English syntax and/or morphology. One might expect that the two systems with morphological features (**clark** and **feat**) would show less difference between English and some of the other languages (all of which have complex morphology) than the other systems. However, although **clark** and **feat** (along with Brown) are the best performing systems overall, they don't show any particular benefit for the morphologically complex languages.[8]

One difference between the Multext-East results and the WSJ results is that on Multext-East, **clark** clearly outperforms all the other systems. This is true for both the English and non-English corpora, despite the similar performance of **clark** and **feat** on (English) WSJ. This suggests that **feat** benefits more from the larger corpus size of WSJ. For the other languages **clark** may be benefiting from somewhat more general morphological features; **feat** currently contains suffix features but no prefix features (although these could be added).

Overall, our experiments on multiple languages support our earlier claim that many of the newer POS induction systems are not as successful as the older methods. Moreover, these experiments underscore the importance of testing unsupervised systems on multiple languages and domains, since both the absolute and relative performance of systems may change on different data sets. Ideally, some of the corpora should be held out as unseen test data if an effective argument is to be made regarding the language- or domain-generality of the system.

5 Learning from induced prototypes

We now introduce a final novel method of evaluating POS induction systems and potentially improving their performance as well. Our idea is based on the prototype-driven learning model of Haghighi and Klein (2006). This model is unsupervised, but requires as input a handful of *prototypes* (canonical examples) for each word class. The system uses a log-linear model with features that include the prototype lists as well as morphological features (the same ones used in **feat**). Using the most frequent words in each gold standard class as prototypes, the authors report 80.5% accuracy (both many-to-one and one-to-one) on WSJ, considerably higher than any of the induction systems seen here. This raises two questions: If we wish to induce prototypes without a tagged corpus or language-specific knowledge, which induction system will provide the best prototypes (i.e., most similar to the gold standard prototypes)? And, can we use the induced prototypes as input to the prototype-driven model (**h&k**) to achieve better performance than the system the prototypes were extracted from?

To explore these questions, we implemented a simple heuristic method for inducing prototypes from the output C of a POS induction system by selecting a few frequent words in each cluster that are the most similar to other words in the cluster and also the most dissimilar to the words in other clusters. For each cluster $c_i \in C$, we retain as candidate prototypes the words whose frequency in c_i is at least 90% as high as the word with the highest frequency (in c_i). This yields about 20-30 candidates from each cluster. For each of these, we compute its average similarity $\mathcal{S}$ to the other candidates in its cluster, and the average dissimilarity $\mathcal{D}$ to the candidates in other clusters. Similarity is computed using the method described by Haghighi and Klein (2006), which uses SVD on word context vectors and cosine similarity. Dissimilarity between a pair of words is computed as one minus the similarity. Finally we compute the average $\mathcal{M} = 0.5(\mathcal{S} + \mathcal{D})$, sort the words by their $\mathcal{M}$ scores, and keep as prototypes the top ten words with $\mathcal{M} > 0.25 * \max_{c_i}(\mathcal{M})$. The cutoff threshold results in some clusters having less than ten prototypes, which is appropriate since some gold standard categories have very few members (e.g., punctuation, determiners).

Using this method, we first tested the various base+proto systems on the WSJ corpus. Results in Table 3 show that the **brown** system produces the best prototypes. Although not as good as using prototypes from the gold standard (**h&k**),

[7] Some results are missing because not all of the corpora were successfully processed by all of the systems.

[8] It can be argued that lemmatization would have given a significant gain to the performance of the systems in these languages. Although lemmatization information was included in the corpus we chose not to use it, maintaining the fully unsupervised nature of this task.

system	many-to-1	1-to-1	vm
brown	**76.1**(8.3)	60.7(10.6)	**68.8**(5.8)
clark	74.5(3.3)	62.1(8.3)	68.6(3.0)
bhmm	71.8(8.6)	56.5(15.0)	65.7(9.5)
vbhmm	68.1(**17.9**)	**67.2**(**20.7**)	67.5(**18.3**)
pr	71.6(9.2)	60.2(17.0)	67.2(12.4)
feat	69.8(-4.1)	52.0(-1.3)	63.1(-4.6)
h&k	80.2	80.2	75.2

Table 3: Scores on WSJ for our prototype-based POS induction system, with prototypes extracted from each of the existing systems [$|C|$:45,$|T|$:45]. Numbers in parentheses are the improvement over the same system without using the prototype step. Scores in bold indicate the best performance (improvement) in each column. **h&k** uses gold standard prototypes.

corpus	**brown**	**clark**
wsj	68.8(5.8)	68.5(3.0)
wsj-s	62.3(2.7)	67.5(3.6)
en	58.5(1.6)	57.9(-3.3)
bg	53.7(2.3)	50.2(-7.1)
cs	49.9(5.0)	48.0(-4.0)
et	45.8(4.9)	44.4(-1.9)
hu	45.8(0.1)	47.0(-5.7)
ro	53.2(0.8)	52.7(-3.3)
sl	51.2(2.9)	51.7(-4.6)
sr	48.0(2.8)	46.4(-4.9)

Table 4: VM scores for **brown+proto** and **clark+proto** on all corpora. Numbers in parentheses indicate improvement over the base systems.

brown+proto yields a large improvement over **brown**, and the highest performance of any system tested so far. In fact, the **brown+proto** scores are, to our knowledge, the best reported results for an unsupervised POS induction system on WSJ.

Next, we evaluated the two best-performing **+proto** systems on Multext-East, as shown in Table 4. We see that **brown** again yields the best prototypes, and again yields improvements when used as **brown+proto** (although the improvements are not as large as those on WSJ). Interestingly, **clark+proto** actually performs worse than **clark** on the multilingual data, showing that although induced prototypes can in principle improve the performance of a system, not all systems will benefit in all situations. This suggests a need for additional investigation to determine what properties of an existing induction system allow it to produce useful prototypes with the current method and/or to develop a specialized system specifically targeted towards inducing useful prototypes.

6 Conclusion

In this paper, we have attempted to provide a more comprehensive review and comparison of evaluation measures and systems for POS induction than has been done before. We pointed out that most of the commonly used evaluation measures are sensitive to the number of induced clusters, and suggested that V-measure (which is less sensitive) should be used as an alternative or in conjunction with the standard measures. With regard to the systems themselves, we found that many of the newer approaches actually perform worse than older methods that are both simpler and faster. The newer systems have introduced potentially important machine learning tools, but are not necessarily better suited to the POS induction task specifically.

Since portability is a distinguishing feature for unsupervised models, we have stressed the importance of testing the systems on corpora that were not used in their development, and especially on different languages. We found that on non-English languages, Clark's (2003) system performed best.

Finally, we introduced the idea of evaluating induction systems based on their ability to produce useful cluster prototypes. We found that the oldest system (Brown et al., 1992) yielded the best prototypes, and that using these prototypes gave state-of-the-art performance on WSJ, as well as improvements on nearly all of the non-English corpora. These promising results suggest a new direction for future research: improving POS induction by developing methods targeted towards extracting better prototypes, rather than focusing on improving clustering of the entire data set.

Acknowledgments

We thank Mark Johnson, Kuzman Ganchev, and Taylor Berg-Kirkpatrick for providing the implementations of their models, as well as Stella Frank, Tom Kwiatkowski, Luke Zettlemoyer and the anonymous reviewers for their comments and suggestions. This work was supported by an EPSRC graduate Fellowship, and by ERC Advanced Fellowship 249520 GRAMPLUS.

References

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.

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.

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

Tomaž Erjavec. 2004. MULTEXT-East Version 3: Multilingual Morphosyntactic Specifications, Lexicons and Corpora. In *Fourth International Conference on Language Resources and Evaluation, LREC'04*, page In print, Paris. ELRA.

Stella Frank, Sharon Goldwater, and Frank Keller. 2009. Evaluating models of syntactic category acquisition without using a gold standard. In *Proceedings of CogSci09*, July.

Kuzman Ganchev, João Graça, Jennifer Gillenwater, and Ben Taskar. 2009. Posterior regularization for structured latent variable models. Technical report, University of Pennsylvania.

Jianfeng Gao and Mark Johnson. 2008. A comparison of bayesian estimators for unsupervised hidden markov model pos taggers. In *Proceedings of EMNLP 2008*, pages 344–352, Morristown, NJ, USA.

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.

Joao Graca, Kuzman Ganchev, Ben Taskar, and Fernando Pereira. 2009. Posterior vs parameter sparsity in latent variable models. In Y. Bengio, D. Schuurmans, J. Lafferty, C. K. I. Williams, and A. Culotta, editors, *Advances in Neural Information Processing Systems 22*, pages 664–672.

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

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.

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

Marina Meilă. 2003. Comparing clusterings by the variation of information. In *Learning Theory and Kernel Machines*, pages 173–187.

B. Merialdo. 1994. Tagging English text with a probabilistic model. *Computational Linguistics*, 20(2):155–172.

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.

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.

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.

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.

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

Jurgen Van Gael, Andreas Vlachos, and Zoubin Ghahramani. 2009. The infinite HMM for unsupervised PoS tagging. In *Proceedings of EMLNP 2009*, pages 678–687, Singapore, August.

Andreas Vlachos, Anna Korhonen, and Zoubin Ghahramani. 2009. Unsupervised and constrained dirichlet process mixture models for verb clustering. In *Proceedings of GEMS 2009*, pages 74–82, Morristown, NJ, USA.