Hierarchical clustering of word class distributions

Grzegorz Chrupała
gchrupala@lsv.uni-saarland.de
Spoken Language Systems
Saarland University

Abstract

We propose an unsupervised approach to POS tagging where first we associate each word type with a probability distribution over word classes using Latent Dirichlet Allocation. Then we create a hierarchical clustering of the word types: we use an agglomerative clustering algorithm where the distance between clusters is defined as the Jensen-Shannon divergence between the probability distributions over classes associated with each word-type. When assigning POS tags, we find the tree leaf most similar to the current word and use the prefix of the path leading to this leaf as the tag. This simple labeler outperforms a baseline based on Brown clusters on 9 out of 10 datasets.

1 Introduction

Unsupervised induction of word categories has been approached from three broad perspectives. First, it is of interest to cognitive scientists who model syntactic category acquisition by children (Redington et al. 1998, Mintz 2003, Parisien et al. 2008, Chrupała and Alishahi 2010), where the primary concern is matching human performance patterns and satisfying cognitively motivated constraints such as incremental learning.

Second, learning categories has been cast as unsupervised part-of-speech tagging task (recent work includes Ravi and Knight (2009), Lee et al. (2010), Lamar et al. (2010), Christodoulopoulos et al. (2011)), and primarily motivated as useful for tagging under-resourced languages.

Finally, learning categories has also been researched from the point of view of feature learning, where the induced categories provide an intermediate level of representation, abstracting away and generalizing over word form features in an NLP application (Brown et al. 1992, Miller et al. 2004, Lin and Wu 2009, Turian et al. 2010, Chrupala 2011, Täckström et al. 2012). The main difference from the part-of-speech setting is that the focus is on evaluating the performance of the learned categories in real tasks rather than on measuring how closely they match gold part-of-speech tags. Some researchers have used both approaches to evaluation.

This difference in evaluation methodology also naturally leads to differing constraints on the nature of the induced representations. For part-of-speech tagging what is needed is a mapping from word tokens to a small set of discrete, atomic labels. For feature learning, there are is no such limitation, and other types of representations have been used, such as low-dimensional continuous vectors learned by neural network language models as in Bengio et al. (2006), Mnih and Hinton (2009), or distributions over word classes learned using Latent Dirichlet Allocation as in Chrupala (2011).

In this paper we propose a simple method of mapping distributions over word classes to a set of discrete labels by hierarchically clustering word class distributions using Jensen-Shannon divergence as a distance metric. This allows us to effectively use the algorithm of Chrupala (2011) and similar ones in settings where using distributions directly is not possible or desirable. Equivalently, our approach can be seen as a generic method to convert a soft clustering to hard clustering while conserving much of the information encoded in the original soft cluster assignments. We evaluate this method on the unsupervised part-of-speech tagging task on ten datasets in nine languages as part of the shared task at the NAACL-HLT 2012 Workshop on Inducing Linguistic Structure.

2 Architecture

Our system consists of the following components (i) a soft word-class induction model (ii) a hierarchical clustering algorithm which builds a tree of word class distributions (iii) a labeler which for each word type finds the leaf in the tree with the most similar word-class distribution and outputs a prefix of the path leading to that leaf.

2.1 Soft word-class model

We use the probabilistic soft word-class model proposed by Chrupala (2011), which is based on Latent Dirichlet Allocation (LDA). LDA was introduced by Blei et al. (2003) and applied to modeling the topic structure in document collections. It is a generative, probabilistic hierarchical Bayesian model which induces a set of latent variables, which correspond to the topics. The topics themselves are multinomial distributions over words.

The generative structure of the LDA model is the following:

$$\begin{aligned} \phi_k &\sim \text{Dirichlet}(\beta), & k &\in [1, K] \\ \theta_d &\sim \text{Dirichlet}(\alpha), & d &\in [1, D] \\ z_{n_d} &\sim \text{Categorical}(\theta_d), & n_d &\in [1, N_d] \\ w_{n_d} &\sim \text{Categorical}(\phi_{z_{n_d}}), & n_d &\in [1, N_d] \end{aligned} \tag{1}$$

Chrupala (2011) interprets the LDA model in terms of word classes as follows: K is the number of classes, D is the number of unique word types, N_d is the number of context features (such as right or left neighbor) associated with word type d, z_{n_d} is the class of word type d in the n_d^{th} context, and w_{n_d} is the n_d^{th} context feature of word type d. Hyperparameters α and β control the sparseness of the vectors θ_d and ϕ_k.

Inference in LDA in general can be performed using either variational EM or Gibbs sampling. Here we use a collapsed Gibbs sampler to estimate two sets of parameters: the θ_d parameters correspond to word class probability distributions given a word type while the ϕ_k correspond to feature distributions given a word class. In the current paper we focus on θ_d which we use to represent a word type d as a distribution over word classes.

Soft word classes are more expressive than hard categories. They make it easy and efficient to express shared ambiguities: Chrupala (2011) gives an example of words used as either first names or surnames, where this shared ambiguity is reflected in the similarity of their word class distributions.

Another important property of soft word classes is that they make it easy to express graded similarity between words types. With hard classes, a pair of words either belong to the same class or to different classes, i.e. similarity is a binary indicator. With soft word classes, we can use standard measures of similarity between probability distributions to judge how similar words are to each other. We take advantage of this feature to build a hierarchical clustering of word types.

2.2 Hierarchical clustering of word types

In some settings, e.g. in the unsupervised part-of-speech tagging scenario, words should be labeled with a small set of discrete labels. The question then arises how to map a probability distribution over word classes corresponding to each word type in the soft word class setting to a discrete label. The most obvious method would be to simply output the highest scoring word class, but this has the disadvantage of discarding much of the information present in the soft labeling.

What we do instead is to create a hierarchical clustering of word types using the Jensen-Shannon (JS) divergence between the word-class distributions as a distance function. JS divergence is an information-theoretic measure of dissimilarity between two probability distributions (Lin 1991). It is defined as follows:

$$JS(P, Q) = \frac{1}{2} \left(D_{\text{KL}}\left(P, M\right) + D_{\text{KL}}\left(Q, M\right) \right) \tag{2}$$

where M is the mean distribution $\frac{P+Q}{2}$ and D_{KL} is the Kullback-Leibler (KL) divergence:

$$D_{\text{KL}}(P, Q) = \sum_i P(i) \log_2 \frac{P(i)}{Q(i)} \tag{3}$$

Unlike KL divergence, JS divergence is symmetric and is defined for any pair of discrete probability distributions over the same domain.

We use a simple agglomerative clustering algorithm to build a tree hierarchy over the word class distributions corresponding to word types (see Algorithm 1). We start with a set of leaf nodes, one for each of D word types, containing the unnormalized word-class probabilities for the corresponding word type: i.e. the co-occurrence counts of word-type and word-class, $n(z, d)$, output by the Gibbs sampler.

We then merge that pair of nodes (P, Q) whose JS divergence is the smallest, remove these two nodes from the set, and add the new merged node with two branches. We proceed in this fashion until we obtain a single root node.

When merging two nodes we sum their co-occurrence count tables: thus the nodes always contain unnormalized probabilities which are normalized only when computing JS scores.

Algorithm 1 Bottom-up clustering of word types

```
$S = \{n(\cdot, d) \mid d \in [1, D]\}$
while $|S| > 1$ do
    $(P, Q) = \mathrm{argmin}_{(P,Q) \in S \times S} \, JS(P, Q)$
    $S \leftarrow S \setminus \{P, Q\} \cup \{\mathrm{merge}(P, Q)\}$
```

The algorithm is simple but not very efficient: if implemented carefully it can be at best quadratic in the number of word types. However, in practice it is unnecessary to run it on more than a few hundred word types which can be done very quickly. In the experiments reported on below we build the tree based only on the 1000 most frequent words.

Figure 1 shows two small fragments of a hierarchy built from 200 most frequent words of the English CHILDES dataset using 10 LDA word classes.

2.3 Tree paths as labels

Once the tree is built, it can be used to assign a label to any word which has an associated word class distribution. In principle, it could be used to perform either type-level or token-level tagging: token-level distributions could be composed from the distributions associated with current word type (θ) and the distributions associated with the current context features (ϕ). Since preliminary experiments with token-level tagging were not successful, here we focus exclusively on type-level tagging.

Given the tree and a word-type paired with a class distribution, we generate a path to a leaf in the tree as follows. If the word is one of the ones used to construct the tree, we simply record the path from the root to the leaf containing this word. If the word is not at any of the leaves (i.e. it is not one of the 1000 most frequent words), we traverse the tree, at each node comparing the JS divergence between the word and the left and right branches, and then descend along the branch for which JS is smaller. We record the path until we reach a leaf node.

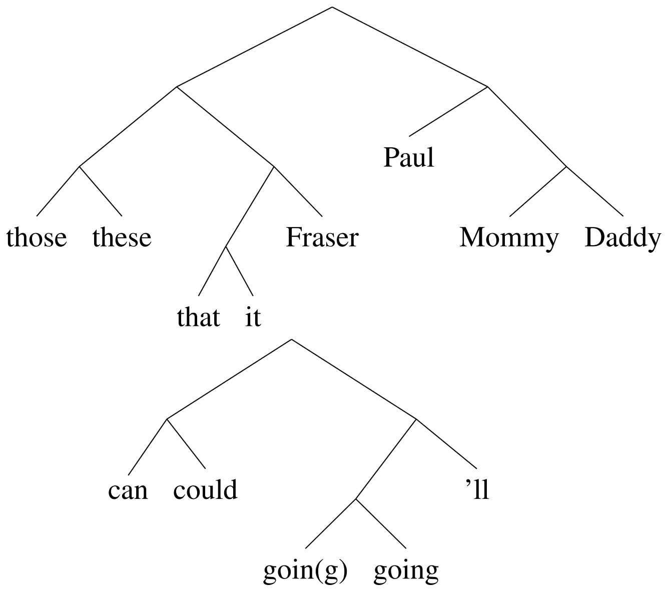

Figure 1: Two fragments of a hierarchy over word class distributions

We can control the granularity of the labeling by varying the length of the prefix of the path from the root to the leaf.

3 Experiments

We evaluate our method on the unsupervised part-of-speech tagging task on ten dataset in nine languages as part of the shared task.

For each dataset we run LDA word class induction[1] on the union of the unlabeled sentences in the train, development and test sets, setting the number of classes $K \in \{10, 20, 40, 80\}$, and build a hierarchy on top of the learned word-class probability distributions as explained above. We then label the development set using path prefixes of length $L \in \{8, 9, \ldots, 20\}$ for each of the trees, and record

[1] We ran 200 Gibbs sampling passes, and set the LDA hyper-parameters to $\alpha = \frac{10}{K}$ and $\beta = 0.1$.

Dataset	K	L	Brown	HCD
Arabic	40	13	39.6	**51.4**
Basque	40	16	39.5	**48.3**
Czech	80	8	42.1	**42.4**
Danish	40	19	50.2	**56.8**
Dutch	40	10	43.3	**54.8**
English CH	10	12	64.1	**67.8**
English PTB	40	8	**61.6**	60.2
Portuguese	80	10	51.7	**52.4**
Slovene	80	19	44.5	**46.6**
Swedish	20	17	51.8	**56.1**

Table 1: Evaluation of coarse-grained POS tagging on test data

Dataset	K	L	Brown	HCD
Arabic	40	13	42.2	**52.9**
Basque	40	16	38.5	**54.4**
Czech	40	19	45.3	**46.8**
Danish	40	20	49.2	**63.6**
Dutch	20	12	49.4	**53.4**
English CH	10	12	66.0	**78.2**
English PTB	80	14	**62.0**	61.3
Portuguese	80	11	52.9	**54.7**
Slovene	80	20	45.8	**51.9**
Swedish	20	17	51.8	**56.1**

Table 2: Evaluation of fine-grained POS tagging on test data

the V-measure (Rosenberg and Hirschberg 2007) against gold part-of-speech tags. We choose the best-performing pair of K and L and use this setting to label the test set. We tune separately for coarse-grained and fine-grained POS tags. Other than using the development set labels to tune these two parameters our system is unsupervised and uses no data other than the sentences in the provided data files.

Table 1 and Table 2 show the best settings for the coarse- and fine-grained POS tagging for all the datasets, and the V-measure scores on the test set achieved by our labeler (HCD for Hierarchy over Class Distributions). Also included are the scores of the official baseline, i.e. labeling with Brown clusters (Brown et al. 1992), with the number of clusters set to match the number of POS tags in each dataset.

The best K stays the same when increasing the granularity in the majority of cases (7 out of 10). On the CHILDES dataset of child-directed speech,

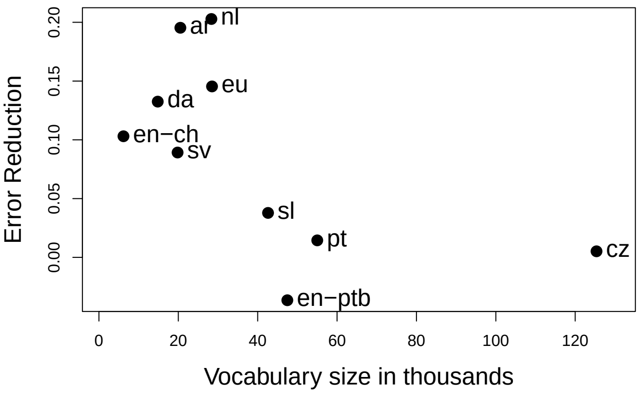

Figure 2: Error reduction as a function of vocabulary size

which has the smallest vocabulary of all, the optimal number of LDA classes is also the smallest (10). As expected, the best path prefix length L is typically larger for the fine-grained labeling.

Our labels outperform the baseline on 9 out of 10 datasets, for both levels of granularity. The only exception is the English Penn Treebank dataset, where the HCD V-measure scores are slightly lower than Brown cluster scores. This may be taken as an illustration of the danger arising if NLP systems are exclusively evaluated on a single dataset: such a dataset may well prove to not be very representative.

Part of the story seems to be that our method tends to outperform the baseline by larger margins on datasets with smaller vocabularies[2]. The scatterplot in Figure 2 illustrates this tendency for coarse-grained POS tagging: Pearson's correlation is -0.6.

4 Conclusion

We have proposed a simple method of converting a set of soft class assignments to a set of discrete labels by building a hierarchical clustering over word-class distributions associated with word types. This allows to use the efficient and effective LDA-based word-class induction method in cases where a hard clustering is required. We have evaluated this

[2] We suspect performance on datasets with large vocabularies could be improved by increasing the number of frequent words used to build the word-type hierarchy; due to time constraints we had to postpone verifying it.

method on the POS tagging task on which our approach outperforms a baseline based on Brown clusters in 9 out of 10 cases, often by a substantial margin.

In future it would be interesting to investigate whether the hierarchy over word-class distributions would also be useful as a source of features in a semi-supervised learning scenario, instead, or in addition to using word-class probabilities as features directly. We would also like to revisit and further investigate the challenging problem of token-level labeling.

References

Bengio, Y., Schwenk, H., Senécal, J., Morin, F., and Gauvain, J. (2006). Neural Probabilistic Language Models. *Innovations in Machine Learning*, pages 137–186.

Blei, D., Ng, A., and Jordan, M. (2003). Latent dirichlet allocation. *The Journal of Machine Learning Research*, 3:993–1022.

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

Christodoulopoulos, C., Goldwater, S., and Steedman, M. (2011). A bayesian mixture model for part-of-speech induction using multiple features. In *EMNLP*.

Chrupala, G. (2011). Efficient induction of probabilistic word classes with LDA. In *IJCNLP*.

Chrupała, G. and Alishahi, A. (2010). Online Entropy-based Model of Lexical Category Acquisition. In *CoNLL*.

Lamar, M., Maron, Y., Johnson, M., and Bienenstock, E. (2010). Svd and clustering for unsupervised pos tagging. In *ACL*.

Lee, Y., Haghighi, A., and Barzilay, R. (2010). Simple type-level unsupervised pos tagging. In *EMNLP*.

Lin, D. and Wu, X. (2009). Phrase clustering for discriminative learning. In *ACL/IJCNLP*.

Lin, J. (1991). Divergence measures based on the shannon entropy. *Information Theory, IEEE Transactions on*, 37(1):145–151.

Miller, S., Guinness, J., and Zamanian, A. (2004). Name tagging with word clusters and discriminative training. In *HLT/NAACL*.

Mintz, T. (2003). Frequent frames as a cue for grammatical categories in child directed speech. *Cognition*, 90(1):91–117.

Mnih, A. and Hinton, G. (2009). A scalable hierarchical distributed language model. In *NIPS*.

Parisien, C., Fazly, A., and Stevenson, S. (2008). An incremental bayesian model for learning syntactic categories. In *CoNLL*.

Ravi, S. and Knight, K. (2009). Minimized models for unsupervised part-of-speech tagging. In *ACL/IJCNLP*.

Redington, M., Crater, N., and Finch, S. (1998). Distributional information: A powerful cue for acquiring syntactic categories. *Cognitive Science: A Multidisciplinary Journal*, 22(4):425–469.

Rosenberg, A. and Hirschberg, J. (2007). V-measure: A conditional entropy-based external cluster evaluation measure. In *EMNLP/CoNLL*.

Turian, J., Ratinov, L., and Bengio, Y. (2010). Word representations: A simple and general method for semi-supervised learning. In *ACL*.

Täckström, O., McDonald, R., and Uszkoreit, J. (2012). Cross-lingual word clusters for direct transfer of linguistic structure. In *NAACL*.