Unsupervised Learning of Bulgarian POS Tags

Derrick Higgins
Educational Testing Service
dchiggin@alumni.uchicago.edu

Abstract

This paper presents an approach to the unsupervised learning of parts of speech which uses both morphological and syntactic information.

While the model is more complex than those which have been employed for unsupervised learning of POS tags in English, which use only syntactic information, the variety of languages in the world requires that we consider morphology as well. In many languages, morphology provides better clues to a word's category than word order.

We present the computational model for POS learning, and present results for applying it to Bulgarian, a Slavic language with relatively free word order and rich morphology.

1 Preliminaries

In designing a model to induce parts of speech (POS categories) from a corpus, the first question which arises is exactly what sort of entities the target categories are. Depending on exactly how these categories are defined, and which words are taken to be members of each, different sorts of linguistic information will clearly be relevant to their identification.

For concreteness, we will be concerned with the part-of-speech categories used in tagging electronic texts such as the Bulgarian Treebank (Simov et al., 2002). Since the goal of this paper is to devise a model which will induce POS categories automatically from an untagged text, with no prior knowledge of the structure of the language, we will be using these tagged corpora as a gold standard to evaluate the performance of competing models.

2 Previous approaches

While this study is unique in attempting to incorporate both syntactic and morphological factors, previous work by other researchers has explored unsupervised methods of deriving clusters of words based on their linguistic behavior.

(Brown et al., 1992) is one of the first works to use statistical methods of distributional analysis to induce clusters of words. These authors define an initial, very fine categorization of the vocabulary of a corpus, in which each word is the sole member of its own category, and then iteratively merge these word classes until the desired level of granularity is achieved. The objective function which they use to determine the optimal set of word classes $\mathcal{C}$ for a corpus is the *inter-class mutual information* between adjacent words in the corpus. Since there is no practical way of determining the classification $\mathcal{C}$ which maximizes this quantity for a given corpus, (Brown et al., 1992) use a greedy algorithm which proceeds from the initial classification, performing the merge which results in the least loss in mutual information at each stage. (Lee, 1997) pursues a similar approach in clustering nouns which occur as direct objects to verbs, but uses a soft clustering algorithm in place of the agglomerative clustering algorithm used by Brown et al., and Lee uses the KL divergence between the nouns' distributions as a measure of closeness, rather than the loss in interclass mutual information. (McMahon and Smith, 1996) employ a similar algorithm to that of Brown et al., but use a top-down search in determining word clusters, rather than a bottom-up one.

A number of other studies have attempted to use distributional analysis to derive POS categories. (Brill et al., 1990) use an ad-hoc similarity metric to cluster words into POS-like classes, but the problem is significantly simplified by their preprocessing of the data to replace infrequent open-

class words with their correct POS tags. Finch & Chater (1994) describe a model based on clustering words in a vector space derived from their corpus contexts, but perform this analysis only for 1,000–2,000 common words in their USENET corpus. Hinrich Schütze (1995) presents perhaps the most sophisticated model of word clustering for POS identification. Schütze first constructs a context vector to represent each word's co-occurrence properties, and then trains a recurrent neural network to predict the word's location in the space based on the context vectors for surrounding words. The output vectors of the network are then clustered to produce POS-like classes. This model architecture, which classifies a word in context, allows the same word to be tagged differently, depending on how it is used.

3 Model components

The approach to the identification of POS categories which we pursue in this paper attempts to incrementally home in on an optimal set of categories through the incorporation of morphological information and local syntactic information. The procedure is uses gradient descent training of a hidden neural network model, including an embedded morphological model based on information from the Linguistica (Goldsmith, 2001) engine for unsupervised morphological analysis. Because this morphological information is the output of a completely unsupervised process of induction, our model of POS category induction is also an unsupervised one.

In the following subsections, we provide a short summary of each of these components of our model of part-of-speech learning, and in Section 4, we present the results of testing our model on a tagged Bulgarian corpus.

3.1 Hidden neural networks

The model which we use for inducing clusters of word tokens in a corpus (which ought to correspond to parts of speech) is actually a generalization of a hidden Markov model, called a hidden neural network (HNN) (Baldi and Chauvin, 1996; Riis, 1998). Each state in the HNN corresponds to a single word cluster, and the category to which a word belongs is determined by Viterbi decoding.

In a hidden neural network, either the transition probabilities, or the observation probabilities, or both, may be replaced with a feed-forward neural network which computes these values on the fly, possibly as a function of information from elsewhere in the observation sequence. This gives an HNN considerably more expressive power than an HMM, because it is not tied strictly to the independence assumptions which are inherent in the architecture of a hidden Markov model.

Gradient descent training of the embedded networks and HNN parameters is entirely straightforward, and is described by (Baldi and Chauvin, 1996).

3.2 Linguistica

While the syntactic information in our model is derived from the state transition and observation parameters through HNN training, the morphological information available to our model of POS category induction is provided by the Linguistica (Goldsmith, 2001) system for unsupervised morphological learning. Linguistica apples a minimum-description length criterion to induce the morphological categories of a language, such as stems and suffixes, from an unlabeled corpus of text from that language. It is important that the morphological analysis which Linguistica provides is not informed by prior knowledge of the language, since this allows our method to remain an *unsupervised* approach to POS learning.

In Goldsmith's framework, a morphological analysis consists of three primary components: a list of stems, a list of suffixes, and a list of *signatures*. A signature, in Goldsmith's conception, is similar to the notion of a morphological paradigm, but more general, and automatically determined from an analysis of a corpus. A stem's signature consists of a list of all of the suffixes which may occur with that stem. Table 1 illustrates some of the highest-ranked signatures found in Linguistica analyses of text from Bulgarian. The occurrence of the string "NULL" in a signature indicates that the stem may also occur with no suffix.

Notable in these signatures is the identification of the gender-markings *-a/-o/-u*.

Table 1: Top-ranked signatures found by Linguistica, for Bulgarian

Signature	Exemplars
а.и.о	азиатск, важн, прав, черн, ...
NULL.та	база, долна,, ... лихба, свобода, ...
NULL.те	бели, контакти,, ... сектори, японски, ...
NULL.то	доколко, рамо,, ... съществуване, широко, ...

3.3 Classification

The morphological analysis provided by Linguistica, however, does not directly make a prediction about a word's part of speech, however. It only tells us whether a word is morphologically simple or complex, and if it is complex, what its stem and suffix are. In order to derive a way of predicting a word's part of speech from this morphological information, we constructed neural network classifiers. The input features for the networks are comprised of the morphological features induced by Linguistica's unsupervised learning algorithm, and the target classes are POS categories of the sort used in tagged corpora.

More precisely, the input features are:

- The word's length, in letters
- The length of the stem, in letters
- The length of the suffix, in letters
- A binary feature indicating whether the word contains punctuation characters
- For each suffix identified by Linguistica, there is a binary feature indicating whether that suffix is used in the word.
- For each suffix identified by Linguistica (including `NULL`), there is a binary feature indicating whether that suffix occurs in the word's signature.

Ultimately, we are interested in the *unsupervised* learning of POS categories, so the training data used by our classifiers will be provided by our hidden Markov model, rather than directly by a tagged corpus. In this section, though, we provide the results of training the classifiers on a gold standard tagged corpus, in order to demonstrate to what degree POS tags are predictable on the basis of morphology alone.

Table 2: Results on prediction of Bulgarian POS tags from morphological information

	Training	Validation	Test
Baseline	21.7%	23.8%	21.1%
Single-layer network	65.6%	66.8%	64.9%
Multi-layer network	68.0%	67.9%	67.9%

Our Bulgarian corpus consists of about 76,000 words of text collected as part of the development of the Bulgarian Treebank project (Simov et al., 2002). The tagset used for this project is very small, consisting of only 11 tags. Bulgarian makes a good test case because it has a high degree of inflection, which necessitates the use of morphological information in determining POS classes, where this might be deemed superfluous in a language like English, which encodes so much through word order.

Table 2 shows the performance of our classifiers on the Bulgarian data set. The classifiers each used 36,000 items for training, 10,000 for validation, and 15,000 as a test set.

4 Learning parts of speech

In this section, we present the results of a modeling experiment designed to assess the usefulness of employing both syntactic and morphological information in the automatic induction of parts of speech from a corpus of text.

In evaluating this model, we will use the measurement of mutual information between the gold standard POS tagging and the model's assignment of induced tags. Since the classes induced by each model will not correspond perfectly to any POS tag found in the target distribution, such as *noun*, *verb*, or *adverb*, we cannot use a simple performance statistic such as "percent correctly classified". The mutual information allows us to measure how closely two distinct distributions match, and is given by Equation 1, where c_1 and c_2 range over the classes in the induced and target classifications, respectively, $\hat{P}(c)$ refers to the empirical frequency of word tokens tagged with category c,

and $\hat{P}(c_1\, c_2)$ refers to the empirical frequency of word tokens tagged as category c_1 by the model, and having category c_2 in the gold standard.

$$I_{\text{eval}} = \sum_{\substack{c_1 \in \mathcal{C}_{\text{induced}}, \\ c_2 \in \mathcal{C}_{\text{target}}}} \hat{P}(c_1\, c_2) \log \frac{\hat{P}(c_1\, c_2)}{\hat{P}(c_1)\hat{P}(c_2)} \quad (1)$$

A high value for the mutual information indicates good agreement between the distributions, and this statistic will tend to zero if the distributions are uncorrelated. The mutual information metric has the advantages that it does not require any human intervention in the evaluation process, and it does not make any assumptions regarding exactly how the two distributions must match up.

(Brill and Marcus, 1992) use a native informant to derive a set of categories from a hierarchical clustering of words, and explicitly label them (as "noun", "verb", etc.). Given this explicit human intervention in the induction procedure (and a certain shuffling of the Penn Treebank category labels), they are able to give evaluation statistics for their method in terms of a simple "percentage correct". Both (Hughes and Atwell, 1994) and (Schütze, 1995) also evaluate their systems by means of a statistic which approximately measures the percentage of words tagged correctly. As in our system, of course, there is no necessary relationship between the induced set of categories and the target categories in the gold standard corpus. However, these authors attempt to avoid this problem by identifying each induced cluster of words with the target category it most often represents. So, for example, if the model induces a class of words which includes many nouns, but only a few verbs, all of the nouns will be counted as correct, and the verbs as incorrect. One difficulty for this method of evaluation is that it is very sensitive to the number of classes induced by a model. In the most extreme case, if every word token is assigned to its own cluster, this evaluation metric will judge that the model has provided a classification which is 100% correct. (Schütze, 1995) simply deals with classifications of words with exactly 200 clusters, so that the question does not arise. This sensitivity of the evaluation metric to the number of clusters induced is also a problem using mutual information, since the mutual information between distributions increases not only with the similarity of the distributions, but also with their entropies. Since we consider induced classifications of only one size, as in Schütze's work, the problem is not crucial in this context.

A shortcoming which is common to all of these approaches using a "percent correct" measurement to evaluate models of part-of-speech induction is that in assigning an induced word cluster to a known target category, such as *noun*, and evaluating the goodness of the cluster according to how well it represents the class *noun*, the assumption is made that it is fine for a target class to be represented by multiple induced clusters, but it is unacceptable for a single induced category to represent a combination of multiple target categories. Of course, this issue does not arise for our approach. Using the mutual information as our evaluation metric, the target and induced tag distributions are treated on a par.

4.1 HNN training

In this section, we present the results of applying a ten-state HNN model of syntactic and morphological factors relating to POS categories to the Bulgarian Treebank corpus.

In Section 3.3, we constructed networks which learned to map words to their POS tags on the basis of morphological information about each word (provided by the Linguistica automatic morphological analysis engine). Using a single-layer or multi-layer perceptron as our model of morphological information, and the parameters of an HNN to represent information about word order, the combined model can be trained using gradient descent. As discussed above, an HNN is distinguished from a hidden Markov model in that certain HMM parameters are replaced by the outputs of feed-forward neural networks. Commonly, the observation probabilities associated with a state are calculated on the fly by a neural network. Thus, while the probability of a string $v_1 \ldots v_n$ according to an HMM is computed as in (2), in this common type of HNN the probability would be calculated as in (3), where the observation probability $b_{s_i, v_{i+1}}$ has been replaced by a function Φ_{s_i} defined by the neural network associated with state

s_i.

$$P_{\text{HMM}}(v_1 \ldots v_n) = \sum_{s_0 \ldots s_n} \pi_{s_0} \prod_{i=0}^{n} a_{s_i,s_{i+1}} b_{s_i,v_{i+1}} \quad (2)$$

$$P_{\text{HNN}}(v_1 \ldots v_n) = \sum_{s_0 \ldots s_n} \pi_{s_0} \prod_{i=0}^{n} a_{s_i,s_{i+1}} \Phi_{s_i}(v_{i+1}) \quad (3)$$

The model introduced in this section differs in two ways from the sort of HNN defined by Equation 3. First, in the general formulation it is perfectly acceptable for the networks Φ_s to be entirely different for each state s in the model. However, in the HNN model of this section, there is only a single neural network used for all ten states. It therefore fulfills the function of a *match network* (Riis, 1998), expressing a probability that the observed symbol is generated by a certain state. Since the network is no longer specific to a certain state, we re-write $\Phi_s(v)$ as $\Phi(s, v)$.

Second, in addition to the match network which uses morphological features to predict the likelihood of a word belonging to a certain class, the model of this section also employs a set of observation probabilities for each state, exactly as in a normal HMM. The idea is that the morphological match networks express broad form-based correlations which hold over parts of speech, for example, that English words with an *-ing* suffix tend to be verbs. The use of observation probabilities in addition to the match networks allows for lexical exceptions to these morphological generalizations. For example, the most common use of the word *icing* is as a noun. With this final addition of a "lexical component" to this section's model of POS induction, the probability equation takes on the form in (4), where $\Phi(s, v)$ is the morphologically-based likelihood of a word belonging to the class expressed by state s, and $b_{s,v}$ is the corresponding lexical likelihood.

$$P_{\text{HNN}}(v_1 \ldots v_n) = \sum_{s_0 \ldots s_n} \pi_{s_0} \prod_{i=0}^{n} a_{s_i,s_{i+1}} \Phi(s_i, v_{i+1}) b_{s_i,v_{i+1}} \quad (4)$$

Exploratory analysis revealed that the clustering algorithm of Brown et al. proved to be useful in defining the initial state of our model, so we use this clustering method to define the initial state of our HNN parameters. The observation probabilities for a given state, representing a certain word class, are determined by the relative frequencies of words belonging to that class (as determined by the algorithm of (Brown et al., 1992)); the probabilities of other words are set to a small initial value. The transition probabilities and initial probabilities for each state are initialized to uniform values. Determining the initial state of the model in this manner guarantees that the HMM's performance will be equal to that of Brown et al.'s clustering method before we begin training, since the two models assign tags to the corpus identically.

We have found that Baum-Welch training of a model initialized in this way is counterproductive over the long run for determining POS categories. However, a small amount of Baum-Welch training can be useful in remedying misclassifications in the model's initial state. For this reason, in the experiments of this section we first train our HMM for ten epochs using the Baum-Welch algorithm, only then incorporating the neural network as our morphological component, and commencing HNN training using gradient descent.

The final piece of information lacking is the initial state of the embedded neural network. For the experiments described here, we used a single-layer perceptron as the morphological component. We could randomly initialize the weights of these models, but that would be detrimental to the initial performance of the model, and training the HNN from that state would be very slow and likely to get caught in local maxima of the corpus likelihood. Therefore, we use the model's initial state to train the initial parameters of the embedded networks off-line. To do this, we present each word token in the corpus as a training example to the network, and using the quickprop algorithm, cause the network to learn a mapping between the morphological features of a word and the class assigned to that word by the network. Since these morphological generalizations are based on the initial categorization provided by the algorithm of (Brown et al., 1992), we hope that they will foster speedy convergence of HNN training.

4.1.1 Results

The results of this combined HNN model, making use of both morphological and syntactic information in constructing linguistic categories, are presented in Figures 1–2 on the following pages.

In Figure 1 we present the evolution of the mutual information metric over the course of training our model on Bulgarian data. The graph represents the change in mutual information between target classifications and induced classifications for the model, which uses a single-layer morphological network. The mutual information begins at about .82 for the model, seeded by the algorithm of Brown et al., and after ten epochs of Baum-Welch training, this sinks to around .785. It then rises to about .83 after morphological information is added, and HNN training improves this value to over .86, showing that the consistency of the induced word clusters with respect to the target tag assignment is improving.

In Figure 2 we present the categories constructed by the same HNN model. For each state in the hidden neural network model, Figure 2 shows a breakdown of the words assigned to that cluster according to the tags they are assigned in the gold standard. Some categories, such as those numbered 1 and 4, are entirely homogeneous. Others show tendencies of different strengths toward different target classes. In any event, the mutual information criterion shows that this clustering, although not ideal, is an improvement over the clusters derived using syntactic information alone.

In sum, adding morphological information to these models of part-of-speech acquisition leads to an increase in performance on Bulgarian data, as measured by our criterion of mutual information. It is somewhat difficult to draw firm conclusions from experiments on a single language, but we hope that this approach will be useful in analyzing other languages which have complex morphology.

5 Future directions

What we hope to have accomplished is a demonstration that the use of morphological information in a model of part-of-speech induction can be of value, and more specifically that the combination of morphological and syntactic generalizations according to the hidden neural network models presented here is a practical way of implementing this approach.

However, the results of the preceding section represent only a first step in exploring how disparate sources of linguistic information may be put to use in deriving POS classes. While we have shown an increase in performance over a purely syntactic baseline model (the algorithm of (Brown et al., 1992)), there are a number of avenues to pursue in extending this work.

First, the quality of the clusters produced will certainly be increased by using a larger training corpus. The corpus used for training our models was on the order of 100,000 words, whereas that used by (Brown et al., 1992) was around 1,000 times this size.

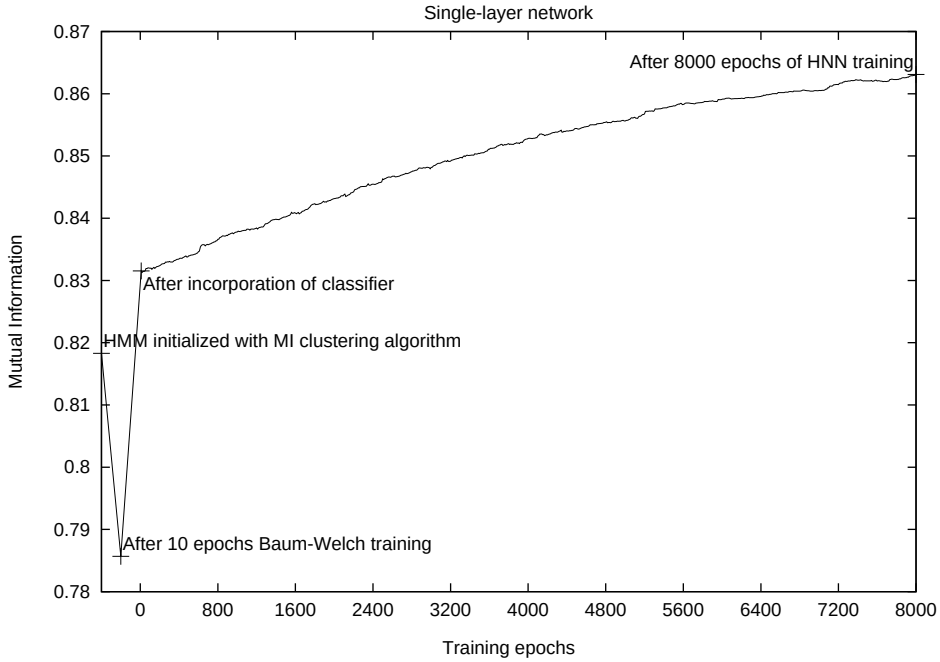
Second, it is worth exploring the parameter of the number of clusters assumed in our model. We have chosen to limit ourselves to ten induced parts of speech, for efficiency of training; however, most previous work in this area has assumed a larger set of clusters.

A third area for improvement is to try different initial syntactic clustering algorithms as a way to seed the HMM on which our HNN model is based. In this article, we used the algorithm of (Brown et al., 1992) to initialize the model. However, this is not the only syntactically-based method for producing word clusters. Since the final state of the model is dependent on the quality of the initial set of clusters, it seems worthwhile to try out other initialization procedures.

Fourth, it is worth investigating whether our approach could be improved upon by attempting to "bootstrap" generalizations from frequent words before attempting to analyze less-frequent ones. After an initial round of HNN training using only high-frequency words, low-frequency words could be added with small initial observation probabilities in each state, and the model could be retrained.

Finally, an important direction for further research is the investigation of a larger set of languages. For our models to be convincing as language-independent ways of acquiring linguistic information, we need to address a broader survey of languages.

Figure 1: Mutual information of Bulgarian word classes induced with respect to the target distributions, over the course of HNN training.



References

Pierre Baldi and Yves Chauvin. 1996. Hybrid modeling, HMM/NN architectures, and protein applications. *Neural Computation*, 8:1541–1565.

Eric Brill and Mitch Marcus. 1992. Tagging an unfamiliar text with minimal human supervision. In *Proceedings of the AAAI Symposium on Probabilistic Approaches to Natural Language*, pages 10–16.

Eric Brill, David Magerman, Mitch Marcus, and Beatrice Santorini. 1990. Deducing linguistic structure from the statistics of large corpora. In *Proceedings of the DARPA Speech and Natural Language Workshop*, pages 275–282.

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.

Steven Finch and Nick Chater. 1994. Distributional bootstrapping: From word class to proto-sentence. In *Proceedings of the Sixteenth Annual Conference of the Cognitive Science Society*, pages 301–306.

John Goldsmith. 2001. Unsupervised learning of the morphology of a natural language. *Computatial Linguistics*, 27(2):153–198.

John Hughes and Eric Atwell. 1994. The automated evaluation of inferred word classifi cations. In *European Conference on Artificial Intelligence*, pages 535–539.

Lillian Lee. 1997. *Similarity-Based Approaches to Natural Language Processing*. Ph.D. thesis, Harvard University, Cambridge, Massachusetts.

John G. McMahon and Francis J. Smith. 1996. Improving statistical language model performance with automatically generated word hierarchies. *Computational Linguistics*, 22(2):217–247.

Søren Kamaric Riis. 1998. *Hidden Markov Models and Neural Networks for Speech Recognition*. Ph.D. thesis, Technical University of Denmark.

Hinrich Sch¨utze. 1995. Distributional part-of-speech tagging. In *Proceedings of the 7th Conference of the European Chapter of the Association for Computational Linguistics (EACL'95)*, pages 141–148.

Kiril Simov, Petya Osenova, Milena Slavcheva, Sia Kolkovska, Elisaveta Balabanova, Dimitar Doikoff, Krassimira Ivanova, Alexander Simov, and Milen Kouylekov. 2002. Building a linguistically interpreted corpus of bulgarian: the bultreebank. In *Proceedings of LREC 2002*, Canary Islands, Spain.

Figure 2: Bulgarian word clusters induced through HNN training (single-layer network)