7
Part-of-Speech Tagging

Part-of-speech tagging is to categorize words according to its part of speech in a given sentence. In Figure 7.1, we present the sentence Flies like flowers with the corresponding part of speech of each word. The task is to give

Word	Flies	like	flowers
POS	noun	verb	noun

Figure 7.1: Part-of-speech tagging example.

each word a tag according to its part of speech. The set of tags, or simply tagset, is fixed within a particular POS tagging task. However, different applications or datasets provide different tagsets, mainly varying POS granularity. For instance, some POS tagsets include only one broad category for verbs, while others include categories like main verb, auxiliary verb, among others. The main difficult of this task is ambiguity, since one word can have different POS tags depending on the context. For instance, the words Flies and flowers can act as verbs in other contexts; and the word like can act as adverb, noun, conjunction, among several other parts of speech. POS provides basic morphological and syntactic information to more complex NLP tasks. It can be even directly used to solve simple information extraction tasks.

We use a general sequence labeling modeling to approach POS tagging. In Section 7.1, we formalize this general task. In order to apply ESL framework to this problem, we still need to define the factorization of the global feature vector $\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y})$ along the output sequence $\boldsymbol{y}$, and the resulting prediction problem. We describe these two aspects in Section 7.2 and Section 7.3, respectively. In Section 7.5, we present empirical results of two ESL applications for POS tagging.

7.1
Task Formalization

The general task of sequence labeling is to find a mapping from an input token sequence $\boldsymbol{x} = (x_1, \ldots, x_N)$ to a label sequence $\boldsymbol{y} = (y_1, \ldots, y_N)$, where $y_t \in S$. That is, each token in $\boldsymbol{x}$ is tagged with a label, or tag, from a given

set S. The output space $\mathcal{Y}(\boldsymbol{x})$ for an input sequence $\boldsymbol{x}$ is the set of all possible sequences of N labels, i.e., $\mathcal{Y}(\boldsymbol{x}) = S^N$. Part-of-speech tagging is an instance of sequence labeling in which S is the given POS tagset.

7.2 Feature Factorization

We use the decomposition scheme from Collins (2002b). Each input token x_t is represented by a vector $\boldsymbol{\Phi}^{\text{surf}}(x_t) = (\phi_1^{\text{surf}}(x_t), \ldots, \phi_M^{\text{surf}}(x_t))$ of M binary features that we call *surface features*. For instance, some surface features that are *present* – have value 1 – in the second token of the example in Figure 7.1 are: `the current word is` like, `the previous word is` Flies, and `the previous word is capitalized`. The number of such features in a dataset with hundreds of thousands of tokens is huge, but just a dozen are active on each token. For a given example $(\boldsymbol{x}, \boldsymbol{y})$, the surface features depend only on the input $\boldsymbol{x}$, which is fixed within the prediction problem. To compose $\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y})$, surface features are combined with the output labels in $\boldsymbol{y}$. Additionally, transition features within $\boldsymbol{y}$ are used to cope with label interdependencies.

Each surface feature $m \in \{1, \ldots, M\}$ is combined with every possible label $s \in S$ to generate the *observation* feature $\phi_{m,s}^{\text{obs}}(x_t, y_t) = \phi_m^{\text{surf}}(x_t) \cdot \mathbf{1}[y_t = s]$, for a given token x_t and its corresponding label y_t. This observation feature indicates whether both the surface feature m is present in token x_t and the token label y_t is equal to s. Then, we can define the observation feature *vector* for a token-label pair (x_t, y_t) as

$$\boldsymbol{\Phi}^{\text{obs}}(x_t, y_t) = \left(\phi_{m,s}^{\text{obs}}(x_t, y_t)\right)_{m \in \{1,\ldots,M\}; s \in S}.$$

Furthermore, we combine these local vectors into the *global* observation feature vector

$$\boldsymbol{\Phi}^{\text{obs}}(\boldsymbol{x}, \boldsymbol{y}) = \sum_{t=1}^{N} \boldsymbol{\Phi}^{\text{obs}}(x_t, y_t),$$

which is the frequency distribution of the observation features in $(\boldsymbol{x}, \boldsymbol{y})$.

For each possible pair of labels $s, r \in S$, the *transition* feature $\phi_{s,r}^{\text{trans}}(y_{t-1}, y_t) = \mathbf{1}[y_{t-1} = s] \cdot \mathbf{1}[y_t = r]$ indicates whether two consecutive labels y_{t-1} and y_t are equal to s and r, respectively. The transition feature vector is then defined as

$$\boldsymbol{\Phi}^{\text{trans}}(y_{t-1}, y_t) = \left(\phi_{s,r}^{\text{trans}}(y_{t-1}, y_t)\right)_{s,r \in S},$$

which is a unit vector whose non-zero position is the one corresponding to

$s = y_{t-1}$ and $r = y_t$. The global transition feature vector is given by

$$\boldsymbol{\Phi}^{\text{trans}}(\boldsymbol{y}) = \sum_{t=2}^{T} \boldsymbol{\Phi}^{\text{trans}}(y_{t-1}, y_t),$$

which is the frequency distribution of the transitions in the output sequence $\boldsymbol{y}$. Finally, the global feature vector is simply the concatenation of the global observation feature vector and the global transition feature vector, that is

$$\boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) = \left(\boldsymbol{\Phi}^{\text{obs}}(\boldsymbol{x}, \boldsymbol{y}), \boldsymbol{\Phi}^{\text{trans}}(\boldsymbol{x}, \boldsymbol{y})\right).$$

7.3 Prediction Problem

The used feature factorization relies on a Markovian property. Thus, in the prediction problem, the best scoring label for a specific token x_t depends only on its label y_t and the previous token label y_{t-1}. In that way, the prediction problem can be reduced to a longest path problem on an weighted directed acyclic graph (DAG), which can be efficiently solved by dynamic programming. In Figure 7.2, we present an example of such DAG for a sentence with three tokens and a tagset with 2 labels (a and b). This graph comprises one layer for

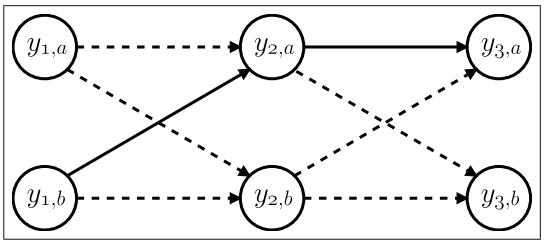

Figure 7.2: Illustrative directed acyclic graph for a sentence $\boldsymbol{x} = (x_1, x_2, x_3)$ and a tagset $S = \{a, b\}$. The continuous path $(y_{1,b}, y_{2,a}, y_{3,a})$ corresponds to the labeling $\boldsymbol{y} = (b, a, a)$.

each token x_t in the input sentence. At every layer t, there is a node labeled $y_{t,s}$ for each label $s \in S$. The node $y_{t,s}$ represents that the t-th token is tagged as s, that is $y_t = s$. For each pair of labels $(s, r) \in S \times S$ and each consecutive layers $t-1$ and t, there is one directed edge $(y_{t-1,s}, y_{t,r})$ in the graph. The edge $(y_{t-1,s}, y_{t,r})$ represents a transition from $y_{t-1} = s$ to $y_t = r$ and its weight is given by

$$s(s, r, x_t) = \langle \boldsymbol{w}^{\text{trans}}, \boldsymbol{\Phi}^{\text{trans}}(s, r) \rangle + \langle \boldsymbol{w}^{\text{obs}}, \boldsymbol{\Phi}^{\text{obs}}(x_t, r) \rangle,$$

where $\boldsymbol{w}^{\text{trans}}$ is the model parameter vector corresponding to transition features and $\boldsymbol{w}^{\text{obs}}$ comprises the parameters for observation features.

Each path from layer $t = 1$ to layer $t = N$ corresponds to a possible output $\boldsymbol{y} = (y_1, \ldots, y_N)$ whose accumulated weight in the graph is

$$s(\boldsymbol{x}, \boldsymbol{y}) = \langle \boldsymbol{w}^{\text{obs}}, \boldsymbol{\Phi}^{\text{obs}}(x_1, y_1) \rangle + \sum_{t=2}^{N} s(y_{t-1}, y_t, x_t). \tag{7-1}$$

By expanding the edge weight function in the formula above, we have that

$$s(\boldsymbol{x}, \boldsymbol{y}) = \sum_{t=1}^{N} \langle \boldsymbol{w}^{\text{obs}}, \boldsymbol{\Phi}^{\text{obs}}(x_t, y_t) \rangle + \sum_{t=2}^{N} \langle \boldsymbol{w}^{\text{trans}}, \boldsymbol{\Phi}^{\text{trans}}(y_{t-1}, y_t) \rangle.$$

And, by using the feature factorization described earlier, we can derive that

$$\begin{aligned} s(\boldsymbol{x}, \boldsymbol{y}) &= \langle \boldsymbol{w}^{\text{obs}}, \boldsymbol{\Phi}^{\text{obs}}(\boldsymbol{x}, \boldsymbol{y}) \rangle + \langle \boldsymbol{w}^{\text{trans}}, \boldsymbol{\Phi}^{\text{trans}}(\boldsymbol{x}, \boldsymbol{y}) \rangle \\ &= \langle \boldsymbol{w}, \boldsymbol{\Phi}(\boldsymbol{x}, \boldsymbol{y}) \rangle, \end{aligned}$$

where $\boldsymbol{w} = (\boldsymbol{w}^{\text{obs}}, \boldsymbol{w}^{\text{trans}})$ is the complete model, that is the concatenation of the observation and transition parameters. Thus, to find the longest path in the aforementioned DAG is equivalent to solve the ESL prediction problem for the presented sequence labeling modeling.

For sequence labeling, we use the loss function $\ell(\boldsymbol{y}, \boldsymbol{y}') = \sum_{t=1}^{T} \mathbf{1}[y_t \neq y'_t]$ that counts the number of mislabeled tokens.

7.4 Basic Features

Our basic features for POS tagging are obtained from dos Santos and Milidiú (2009a). We use the following features:

- *Word:* The surface form of a token;
- *Prefix/Suffix:* Word prefixes and suffixes up to 5-character long;
- *Known Word Prefix:* Adding (or subtracting) 5-character prefix (or suffix) results in a known word, where known words are the ones that occur in the training dataset;
- *Known Word Bigram:* Occurrence of the word before (or after) a specific word in a given long list of word bigrams. For instance, for the English language, if the word appears after `to`, then it is likely to be a verb in the infinitive form;
- *Word Window:* Words of the previous two tokens and the next two tokens.

7.5 Empirical Results

We evaluate our system performances on two POS datasets: Mac-Morpho (Aluísio et al., 2003), a Portuguese language corpus; and Brown (Francis and Kučera, 1982), an English language corpus. In Table 7.1, we present some statistics of these datasets. Both datasets are split into training and test

Dataset	Language	Tagset size	Training Tokens	Test Tokens
Mac-Morpho	Portuguese	22	1,007,671	213,794
Brown	English	182	950,975	210,217

Table 7.1: Basic statistics of the part-of-speech tagging datasets.

partitions. Brown and Mac-Morpho have relatively the same size, but Brown includes a much larger tagset. Performance on POS tagging is reported on simple token accuracy, that is the percentage of correctly tagged tokens among all tokens.

7.5.1 Mac-Morpho Dataset

The best performing system on the Mac-Morpho dataset is the ETL Committee (dos Santos and Milidiú, 2009a), an ensemble composed by 100 ETL models. We compare our system to this ensemble system and also to the best single model ETL-based system. In Table 7.2, we depict the performance of these systems. We notice that ESL reduces the accuracy error by 5.9% when

System	Accuracy
ETL single model	96.75
ETL Committee	96.94
ESL	**97.12**

Table 7.2: Performances on the Mac-Morpho dataset.

compared to ETL Committee and by 11.4% when compared to the single model ETL system. These are substantial improvements on this dataset, since there is not much room for improvements.

Regarding ESL training meta-parameters, we use the following setting. The number of epochs is 50 and the loss weight parameter C is set to 50. One epoch corresponds to one complete pass over all examples in the training set. The minimum and the maximum feature template length is set to 2.

7.5.2 Brown Dataset

The best performing system on the Brown dataset is also ETL Committee. Thus, in Table 7.3, we again present the performances of the best single model ETL system, ETL Committee, and ESL. We notice that

System	Accuracy
ETL single model	96.69
ETL Committee	**96.83**
ESL	96.72

Table 7.3: Performances on the Brown dataset.

ESL outperforms the single model ETL system, but does not outperform ETL Committee. ETL Committee error is 3.4% smaller than ESL error. Nevertheless, ESL is still competitive with state-of-the-art systems. Moreover, we can also train ensembles of ESL models and probably have some gain in performance.

We use the following values for ESL meta-parameters. The number of epochs is 50 and the loss weight parameter C is set to 100. One epoch corresponds to one complete pass over all examples in the training set. The minimum and the maximum feature template length is set to 2.